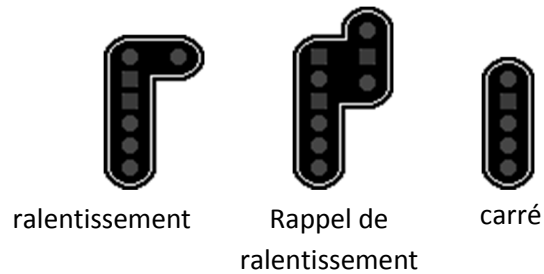


## Notice d'utilisation du programme signaux\_complexes\_GL

Cette notice montre comment utiliser le programme signaux\_complexes\_GL pour utiliser des signaux complexes français avec ou sans CDM rail.

Exemples de signaux complexes :



Rappel de  
ralentissement combiné à  
un avertissement

**Matériel nécessaire :**

- Le programme « signaux\_complexes\_GL »
- CDM rail V5.1 mini.
- décodeur de feux « led dekoder » de **DigitalBahn** équipé de son logiciel « led\_signal\_10 » ou un décodeur **CDF** ou **LEB** ou **LDT-DEC-SNCF** ou **Leb-Modélisme** ou **UniSemaf** ou **SR**

<b>Introduction</b>	<b>4</b>
Fonctionnement avec CDM rail et signaux_complexes_GL	4
Fonctionnement de Signaux_complexes_GL en autonome	5
<b>Installation</b>	<b>5</b>
Pare feu	6
Exclusion de sécurité	6
Refus de modification des fichiers du dossier par windows ou d'exécution	6
Windows 10 et 11	6
<b>Fonctionnement</b>	<b>8</b>
Différences entre cantons et zones de détections	9
Signaux complexes virtuels et réels	9
Implantation avec une zone de détection par canton sur des détecteurs de pleine voie :	10
Restrictions et spécificités du programme client pour les signaux	11
<b>Modélisation du réseau</b>	<b>11</b>
<b>Configuration générale</b>	<b>11</b>
<b>Onglet CDM Rail</b>	<b>12</b>
<b>Onglet Mode autonome</b>	<b>14</b>
A propos des connexions par réseau Ethernet ou wifi aux centrales (uniquement pour le mode autonome):	15
<b>Principes généraux de modélisation</b>	<b>16</b>
<b>Modification des aiguillages</b>	<b>17</b>
Suppression simultanée d'aiguillages	18
Modélisation des TJD	19
Modélisation des croisements	21
Modélisation des aiguillages triples	23
<b>Modélisation des TJD et des TJS</b>	<b>25</b>
Codification d'une TJS	26
Modélisation d'un buttoir sur un aiguillage	27
<b>Modélisation des signaux</b>	<b>28</b>
Signaux directionnels	30
Feux pour plusieurs voies simultanées	31
Spécificités du décodeur Unisemaf	32
Spécificités du décodeur Stéphane Ravaut (SR)	34
Autres paramètres modifiables des signaux	35
Suppression simultanée de signaux	35
<b>Modélisation des branches</b>	<b>36</b>
<b>Modification évènements actionneurs/détecteurs</b>	<b>38</b>
Déclencher une fonction F d'une locomotive (F0 à F12 ou F28) , accessoire ou son	39
Déclencheur "zone de détection"	41

Actionner un passage à niveau à une ou plusieurs voies	41
<b>Erreurs à la lecture du fichier de configuration</b>	<b>44</b>
<b>Erreurs à l'exécution</b>	<b>44</b>
<b>Interfaces XpressNet</b>	<b>45</b>
<b>TCO (Tableau de commande optique)</b>	<b>46</b>
<b>Information sur les décodeurs d'accessoires en DCC</b>	<b>49</b>
<b>Utilisation du programme signaux_complexes_GL avec CDM rail</b>	<b>50</b>
<b>Fenêtre DEBUG</b>	<b>51</b>
<b>Pilotage individuel des signaux</b>	<b>53</b>
Vérification du pilotage des décodeurs	53
<b>Pilotage des vitesses et des fonctions F des trains</b>	<b>54</b>
<b>Simulateur</b>	<b>54</b>
<b>Lecture / Ecriture de variables de configuration (CV)</b>	<b>55</b>
Ecriture d'un CV seul dans un accessoire	55
Lecture d'un CV seul depuis un accessoire	55
Ecriture de CV en cascade dans un accessoire depuis un fichier	55
<b>Lire un fichier de trames CDM</b>	<b>56</b>
Création du fichier trames	56
Lecture du fichier trames CDM	56
<b>Arrêt des logiciels</b>	<b>57</b>
<b>Annexe - Liste des états d'un signal SNCF complet à 10 feux</b>	<b>57</b>

## Introduction

Signaux\_complexes\_GL a été écrit en Delphi7 enrichi d'un OCX pour la communication série et USB, et de la bibliothèque sockets. Pour les programmeurs, il existe une notice avancée qui décrit la programmation pour l'aide à la modification du programme signaux complexes

Le programme localise les trains en circulation sur le réseau à partir des événements des détecteurs de zone transmis soit directement par la centrale via le bus XpressNet de l'interface (sans CDM Rail) soit transmis par CDM Rail. Ce programme gère :

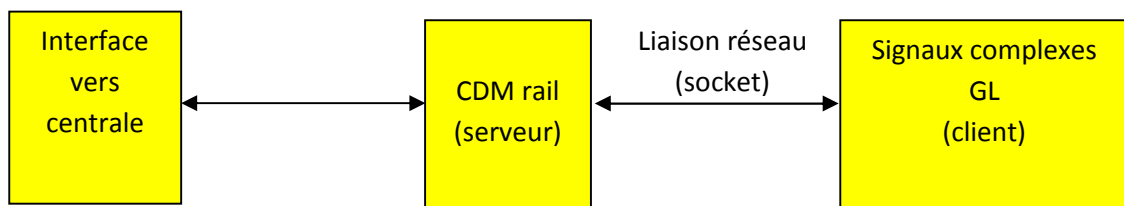
- les signaux complexes
- les actionneurs évolués
- le mode TCO
- le lancement de CDM rail en automatique avec le LAY (sauf Win 10 et 11) désiré ainsi que de l'interface.

Le programme signaux\_complexes\_GL utilise un fichier de configuration (*configgenerale.cfg*) qui contient des paramètres et les réglages à configurer. Les paramètres sont accessibles depuis la fenêtre de configuration générale. Ils sont décrits dans cette notice.

Signaux\_complexes\_GL a deux modes de fonctionnement :

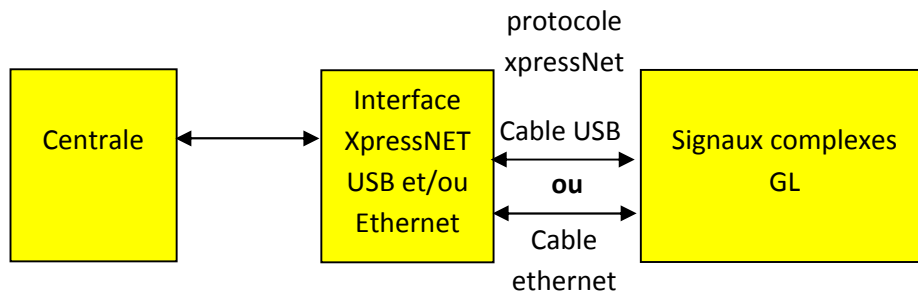
### Fonctionnement avec CDM rail et signaux\_complexes\_GL

Dans ce mode, Signaux\_Complexes\_GL est client de CDM rail pour piloter les signaux complexes. Il peut être exécuté sur le même ordinateur que CDM rail, ou un ordinateur différent. Dans ce cas ces deux ordinateurs seront reliés via une liaison réseau, et il est nécessaire de renseigner l'adresse IP V4 du PC exécutant CDM rail dans les paramètres du mode « CDM rail ».



## Fonctionnement de Signaux\_complexes\_GL en autonome

Dans ce mode, Signaux\_complexes\_GL est autonome et il n'a pas besoin de CDM rail pour piloter les accessoires du réseau. Cette fonctionnalité est valide uniquement avec les centrales disposant d'une connexion Xpressnet V3.5. Dans ce cas, Signaux\_complexes\_GL doit être relié à la centrale par le bus XpressNet (via une interface XpressNET-USB ou XpressNet-USB-Ethernet ou Genli ou centrale comportant nativement de XpressNet [Digikeijs DR5000, Roco...]). Ce mode permet de piloter les locomotives en mode « raquette » et le programme se charge de piloter les signaux complexes. En aucun cas il ne gère la sécurité des convois. Les paramètres se trouvent dans l'onglet « mode autonome » de la fenêtre de configuration générale.



Il existe un mode hybride, qui permet d'utiliser CDM d'une part et le programme Signaux\_complexes\_GL de façon autonome, c'est-à-dire sans liaison client-serveur entre les deux programmes. Dans ce cas, il faut mettre 0 dans la variable « Adresse IP du PC exécutant CDM rail ». CDM rail communique avec l'interface par la liaison USB et Signaux\_complexes\_GL communique avec l'interface en Ethernet. Ce mode nécessite une interface XpressNet Ethernet.

## Installation

Il est indispensable de procéder à l'installation du logiciel. Pour cela, faire un clic droit sur le fichier `Installeur.exe` et sélectionner « Exécuter en tant qu'administrateur ».

L'installation va installer un composant nécessaire à la communication USB avec la centrale ainsi que ses autorisations dans le registre. Signaux\_complexes\_GL ne pourra pas se lancer si ce composant (`mscomm32.ocx`) n'est pas installé.

Pour une mise à jour, il n'est plus nécessaire de procéder à une installation, il suffit seulement de copier le fichier exécutable (`signaux_complexes_GL.exe`) dans votre répertoire de travail.

A l'exécution de Signaux\_complexes\_GL, si votre PC est connecté à Internet, une vérification d'une nouvelle version est lancée et il vous sera proposé de la télécharger. Si vous choisissez oui, le fichier zip sera réceptionné dans le répertoire téléchargements de votre PC

## **Pare feu**

Pour pouvoir utiliser la liaison socket entre CDM rail et le logiciel signaux\_complexes\_GL, si le pare feu windows ou de votre antivirus est activé, il faut leur ajouter une exception pour CDM rail ET signaux\_complexes\_GL (Dans le pare feu windows : *autoriser une application ou une fonctionnalité via le pare feu windows*), cliquer sur autoriser un autre programme, chercher CDM rail puis dans un deuxième temps, signaux\_complexes\_GL.exe. Le port à autoriser est le 9999 (par défaut).

## **Exclusion de sécurité**

Il est possible que Windows defender détecte par abus un trojan dans Signaux\_complexes\_GL.exe. Dans ce cas il faut ajouter une exclusion de sécurité en suivant le procédé décrit à ce lien :

<https://support.microsoft.com/fr-fr/help/4028485/windows-10-add-an-exclusion-to-windows-security>

Vous pouvez toujours faire scanner le fichier pour le vérifier sur un site antivirus comme *Kaspersky* par exemple :

<https://opentip.kaspersky.com/>

## **Refus de modification des fichiers du dossier par windows ou d'exécution**

Il est possible qu'en cas de modification des fichiers de configuration, un refus d'accès soit affiché par windows (W10). Il s'agit d'un problème de droit administrateur. Pour remédier à cela suivre la manipulation suivante à faire en mode administrateur. Faire clic droit sur le répertoire de signaux\_complexes\_GL, puis sélectionner propriétés et ensuite l'onglet sécurité.

Cliquer sur le bouton modifier au milieu.

Dans la sélection supérieure, choisir le profil (ex *tous les utilisateurs* ou *administrateur* ou le nom de votre session windows).

En bas, cliquer sur la ligne contrôle total puis cliquer sur autoriser et ok puis encore ok. Les autorisations totales ont été déclarées pour les profils choisis.

Vous pouvez appliquer la même règle pour tous les profils.

## **Windows 10 et 11**

Avec Windows 10 et 11, si on copie les fichiers de signaux\_complexes dans le répertoire "programmes", il est nécessaire d'activer le mode de compatibilité Windows XP. Faire un clic droit sur le fichier

"signaux\_complexes\_GL.exe" et cliquer sur propriétés, onglet compatibilité, cliquer sur "Lancer la résolution des problèmes de compatibilité", il devrait proposer la compatibilité pour windows XP.

Avec Windows 11 il est vivement conseillé de désactiver l'assistant de concentration (focus assist) : Ces paramètres se trouvent dans les paramètres windows : système / Assistant de concentration : désactiver l'assistant de concentration ; dans règles automatiques, désactiver toutes les options. La désactivation de l'assistant de concentration évite à la barre des tâches de clignoter pendant les traitements graphiques.

## Fonctionnement

Les signaux gérés de base par CDM rail sont de trois types : carré violet, canton 3 feux ou 4 feux. Néanmoins, CDM rail gère ces feux de la même façon. Pour pouvoir utiliser les feux complexes, il faut utiliser un décodeur externe, piloté par le programme signaux complexes. Ce programme client communique avec le programme serveur (CDM Rail).

Les décodeurs des signaux complexes étant pilotés par le même bus que les aiguillages, le plan d'adresses des signaux complexes ne doit pas interférer avec les adresses des feux CDM, aiguillages ou des actionneurs.

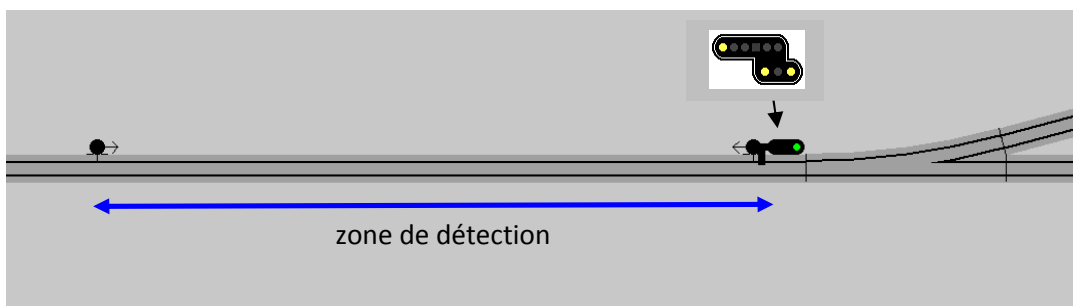
**Les feux étant pilotés par signaux\_complexes, il ne faut pas mettre d'adresse de signal dans CDM rail. Laisser le champ vide. On peut aussi désélectionner "autoriser le pilotage des signaux" dans les options de RUN dans CDM rail.**

Les signaux complexes gérés par le programme client n'ont rien à voir avec les signaux déclarés dans le réseau dans CDM rail. La représentation de la signalisation étant obligatoire sous CDM rail, elle ne correspond pas forcément à des signaux physiques installés ; CDM rail ne pilotant pas la signalisation complexe (étendue). Les deux processus sont donc dissociés.

Les signaux complexes du programmes client sont implantables « librement » conformément aux règles de la signalisation française. Ils ne sont pas liés aux signaux de CDM rail. Cela signifie que l'on aura d'une part les signaux implantés dans CDM rail et d'autre part les signaux implantés dans le programme client qui représentent les signaux complexes sur le réseau. A un endroit particulier du réseau on pourra donc avoir un signal CDM et un signal complexe physiquement implanté sur le réseau ou pas de signal complexe.

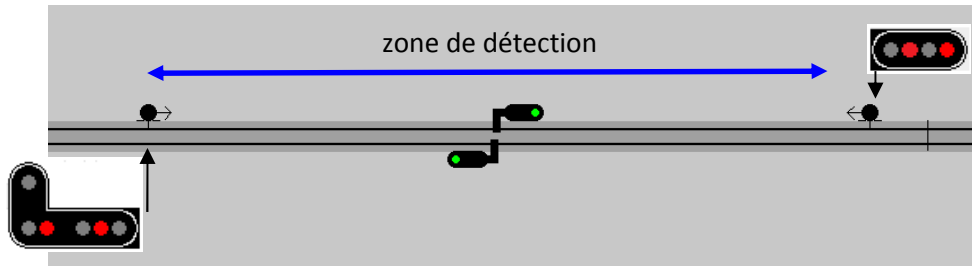
Les signaux complexes doivent être installés en fin de zone de détection, avant un aiguillage ou un grill pour annoncer le rappel de ralentissement ou le carré (et dans ce cas le signal précédent doit être un signal ralentissement).

Exemple :



Un signal complexe est toujours associé à la fin d'une zone de détection (comme ci-dessus), ou ci-dessous dans le cas d'une zone de détection de pleine voie :



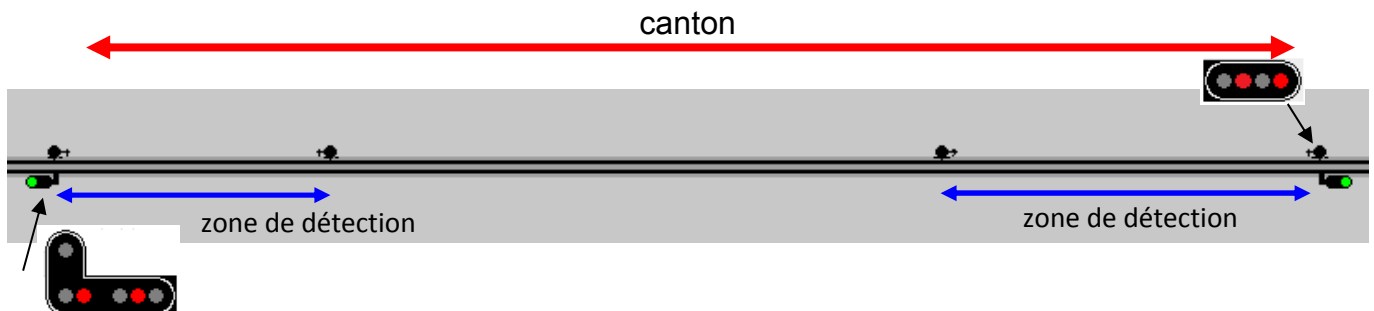


Le signal complexe est bien sûr sensible au sens de circulation de la voie sur lequel il est implanté.

## Différences entre cantons et zones de détections

Une zone de détection est un équipement électrique qui permet la détection d'un train sur le réseau. Un canton peut être constitué de plusieurs zones de détections obligatoirement contiguës. Un canton est délimité par des signaux (dits signaux de cantonnement) dans lequel un seul train est autorisé. Deux signaux complexes sont donc positionnés aux limites d'un canton dans lequel un seul train peut être présent. CDM rail conseille de mettre deux zones de détection par canton en regard de chaque signal d'extrémité, mais CDM rail fonctionne aussi avec une seule zone par canton.

Implantation avec deux zones de détection par canton :

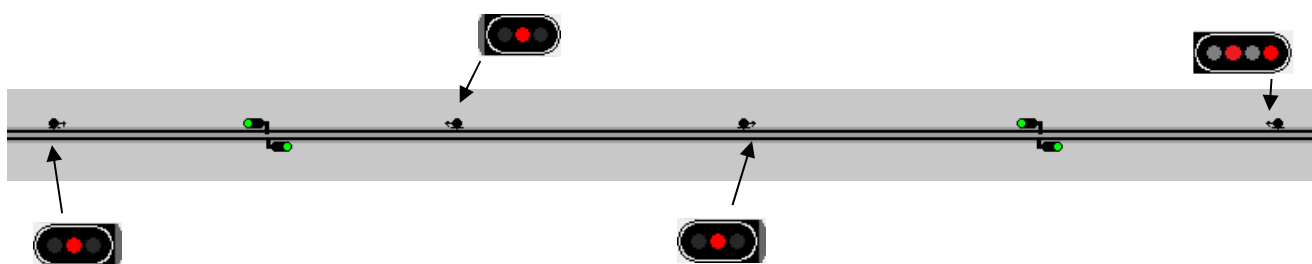


Les signaux complexes sont bien sûr implantés en limite de canton.

## Signaux complexes virtuels et réels

Il n'est pas obligatoire *d'implanter* un signal complexe si l'on n'en dispose pas (par exemple pour des questions de coûts). Néanmoins pour des raisons de fonctionnement, il **faut** « implanter » ou plutôt déclarer à la place un signal virtuel (il possèdera une adresse sur le réseau mais il ne sera pas piloté). Ceci permet par exemple de piloter un signal de ralentissement malgré que l'on ne soit pas en possession de son homologue, le signal de rappel de ralentissement. En effet le programme signaux\_complexes va tester le signal suivant pour connaître son état. Il est donc nécessaire de gérer un signal virtuel. Un signal virtuel a un décodeur sélectionné sur "*rien*" dans la configuration.

## Implantation avec une zone de détection par canton sur des détecteurs de pleine voie :



Pour les signaux complexes, le canton est encadré de deux signaux (cas d'une voie banalisée pour la circulation dans les deux sens) ou d'un seul signal si la circulation se fait toujours dans le même sens.

Les signaux complexes seront implantés sur les fins de zone de détection dans le sens de la circulation, car le changement des signaux complexes se fait sur la retombée de la zone de détection, dans le sens de circulation (front descendant)

Le programme client pilote les signaux complexes du réseau. Il doit donc connaître le réseau. Pour cela, il est nécessaire de renseigner les configurations "branches", "aiguillages" et "signaux" du panneau de configuration.

## **Restrictions et spécificités du programme client pour les signaux**

### **En utilisation avec CDM rail en mode RUN sans itinéraire :**

Les signaux seront positionnés en fonction des aiguillages. C'est à l'opérateur de manœuvrer les aiguillages suffisamment à l'avance avant le passage du train pour que la présentation des signaux soit cohérente.

D'autre part il faut manœuvrer les aiguillages après le passage du train sur le détecteur suivant, sinon la route sera mal évaluée ; elle sera resynchronisée plus tard, mais il aura création d'un train « fantôme » : le nombre de trains affichés sera supérieur.

### **En utilisation avec CDM rail en mode RUN avec itinéraire(s) :**

Les aiguillages sont positionnés 1 canton avant le passage du train, ce qui est trop tard pour l'affichage d'un signal de ralentissement.

Pour une rapidité d'exécution optimale, dans CDM, il faut absolument dévalider les deux options de création des logs dans les menus « Comm IP/créer un fichier de log » et « Autoriser le log des événements de service » (avant de lancer le serveur de l'interface). Cette opération n'est à faire qu'une seule fois.

Le programme signal client gère les signaux violet, blanc, vert, rouge, carré, sémaphore, jaune, jaune clignotant, ralentissement 30 ou 60 et rappel de ralentissement 30 ou 60.

Pour afficher les signaux vert clignotant, rouge clignotant et blanc clignotant, il faut les programmer spécifiquement le programme par Delphi 7.

## **Modélisation du réseau**

La modélisation du réseau est sa description. On en crée sa description depuis le panneau de configuration.

## **Configuration générale**

Le panneau de configuration générale est accessible depuis le menu « Divers / configuration générale ». Si vous changez une option et si vous cliquez sur le bouton « Enregistrer la configuration et fermer », la modification sera apportée au fichier de configuration. L'option de lancement du LAY avec CDM Rail ne fonctionne qu'avec CDM Rail 7.03 minimum. Le lancement du serveur IP de CDM rail et le démarrage de l'interface sont automatiques si vous sélectionnez l'option correspondante. Il est possible de modifier la définition, des signaux et des actionneurs, des aiguillages (ou des TJD) et des branches. On peut aussi ajouter ou supprimer des éléments.

## Onglet CDM Rail

**Configuration générale**

CDM Rail | Mode autonome | Aiguillages | Branches | Signaux | Actionneurs/Détecteurs

**CDM Rail**

Adresse IP du serveur CDM rail : 127.0.0.1  
Port du serveur CDM rail : 9999

**Au démarrage de Signaux\_Complexes**

☐ Vérifications de nouvelle version au démarrage  
☐ Information sur la version actuelle  
☒ Avec TCO ☐ Bandeau TCO masqué  
☐ Fenêtre étendue  
☐ Lancer CDM Rail au démarrage

Nom du fichier LAY à ouvrir au démarrage de CDM rail (nécessite la version mini 7.03 de CDM Rail)  
RESEAU\_GILY\_SIGNAL\_AJOUTE.LAY

**Services CommIP CDM Rail**

Liste des événements demandés à CDM Rail:

☒ Aiguillages ☐ Signaux (non utilisé)  
☒ Détecteurs  
☒ Actionneurs  
☐ Position des trains (non utilisé)

**Au démarrage de CDM Rail : serveur**

☐ Ne pas démarrer de serveur  
☒ XpressNet (lenz.roc)  
☐ P50X intellibox  
☐ Sprog  
☐ HSI88/HSI88-USB  
☐ ECOS ESU  
☐ RS2PC (Rs FeedBack interface)  
☐ FIS88 FeedBack (S88/S88N)  
☐ DCC++

**Au démarrage de CDM Rail : interface LENZ**

☒ Automatique  
☐ LI-USB  
☐ LI-101F  
☐ LI-100F  
☐ LI-100  
☐ Genli

**Divers**

Seuil du nombre de détecteurs trop distants : 3  
Taille de la fonte de la fenêtre principale : 12

Paramètres de connexion et d'échange avec CDM rail

Fermer sans enregistrer la configuration | Enregistre la configuration et Fermer

**Signaux complexes GL**

### Adresse IP du serveur CDM Rail / Port du serveur

Il s'agit du PC sur lequel CDM fonctionne. Signaux\_complexes\_GL utilise un socket (adresse ip : port) pour communiquer avec le serveur de CDM rail. Les deux programmes peuvent donc être sur deux PC différents. L'adresse 127.0.0.1 représente le PC sur lequel le programme Signaux\_complexes\_GL s'exécute, lorsque les deux programmes s'exécutent sur le même PC, et c'est donc cette adresse ip qu'il faudra utiliser.

Le port doit être le même que celui de CDM rail (9999 par défaut) ; modifiable dans CDM rail dans le menu Comm.IP / Paramétrage de la liaison IP. Signaux\_complexes\_gl ne gère pas de nom d'hôte.

Il est possible de vérifier ou non si une nouvelle version du programme est disponible avec les coches "**Vérification de nouvelle version au démarrage**" et "**Information sur la version actuelle**".

**Fenêtre étendue** permet de lancer signaux complexes en plein écran ou en fenêtre réduite.

**Avec TCO** permet d'utiliser le programme avec un TCO (tableau de contrôle optique) que vous aurez préalablement dessiné.

**Bandeau TCO masqué** : masque le bandeau de configuration du TCO au démarrage.

**Seuil du nombre de détecteurs trop distants:** lors d'une recherche de deux détecteurs non consécutifs, c'est la limite maximale du nombre de détecteurs considérés comme trop distants. Cette valeur est par défaut à 3 pour les petits et moyens réseaux. Elle peut être portée à 4 ou 5 pour les grands réseaux qui ont beaucoup de zones de détection consécutives entre les signaux.

**Lancer CDM Rail au démarrage :** permet de lancer automatiquement CDM rail avec le réseau de votre choix au démarrage du programme signaux\_complexes\_GL. Dans ce cas, le serveur de CDM est automatiquement ouvert, et la liaison socket est établie.

*Note : au premier chargement du LAY, il faudra changer le chemin d'accès depuis CDM Rail. La sélection du répertoire de stockage du LAY sera alors mémorisée dans CDM.*

Il est possible de laisser le champ du LAY vide. Dans ce cas CDM s'ouvre avec le dernier LAY. Utiliser cette option si l'ouverture du LAY ne fonctionne pas (rencontré en Windows 10 et 11).

Choisissez alors le nom du serveur et le type d'interface dans la partie droite de la fenêtre.

**Taille de fonte de la fenêtre principale :** ne fonctionne que si signaux\_complexes a été compilé avec RAD\_Studio, ce qui n'est pas le cas de la version distribuée.

## Onglet Mode autonome

**Configuration générale**

CDM Rail | **Mode autonome** | Aiguillages | Branches | Signaux | Actionneurs/Détecteurs

Accès USB/Série à l'interface vers la centrale via XpressNet

1. Protocole série USB / COM : COM5:57600,N,8,1,2

2. Temporisation d'envoi des octets de la trame (ms) : 50

3. Temporisation d'attente de la réponse de l'interface (x 100 ms) : 7

4. Entête des trames vers l'interface

☐ 0 : Sans entête (interfaces série)

☒ 1 : Entête FF FE (interfaces natives USB)

☐ 2 : Entête E4 - Suffixe 0D 0D 0A (arduino XpressNet)

Accès Ethernet à l'interface vers la centrale via XpressNet

Adresse IP interface : 192.168.1.23

Port de l'interface : 5550

Au démarrage de signaux complexes en mode autonome

☒ Initialisation des aiguillages

Temporisation de séquençement d'init (ms) : 80

☐ Demande positions des aiguillages à la centrale

☒ Connexion de l'interface Xpressnet en COM ou USB

☒ Connexion de l'interface Xpressnet en Ethernet

Ces paramètres sont utilisés en fonctionnement sans CDM Rail

Fermer sans enregistrer la configuration | Enregistre la configuration et Fermer | Enregistre la configuration

**Notes explicatives :**

1. Port COM de l'adresse USB de l'interface XpressNet. Attention de COM1 à 9 - Si le port de l'interface USB>9, il faut le changer manuellement dans le gestionnaire des périphériques. Si COMX : Signaux complexes détecte le port automatiquement (mais le démarrage est plus long)
2. Valeur de temporisation entre deux octets transférés à l'interface. Elle peut être nulle. Pour les interfaces série sans protocole (0) comme le GENLI, il est conseillé de la positionner à une valeur de l'ordre de 30 (ms). Pour les interfaces avec protocole matériel RTS-CTS (2) cette variable est ignorée.
3. Valeur maximale par tranche de 100 ms qui définit le temps d'attente de la réponse de l'interface après une trame qui lui est transférée. Cette valeur est à tester en fonction de votre interface. En cas de dépassement de la valeur, un message « pas de réponse de l'interface » sera affiché.
4. Pour l'utilisation d'interfaces série (GENLI), cette valeur doit être à 0. Pour les interfaces utilisant nativement de l'USB, cette valeur doit être à 1. La valeur 2 est utilisée exclusivement pour des interfaces à base d'arduino pour xpressnet.

On peut utiliser Signaux\_complexes\_GL de façon autonome. Cet onglet décrit les méthodes de connexion à l'interface (USB ou Ethernet).

**Protocole série USB/COM** décrit le port USB ou COM pour qu'il se connecte à l'interface XpressNet. Cette ligne contient le numéro de port de l'interface et le protocole à utiliser. Ce numéro de port va de 1 à 9. On peut utiliser **COMX** pour détecter automatiquement le port de l'interface (de 1 à 9). Dans ce cas le démarrage est plus long.

Si le port de l'interface USB>9, il faut le changer manuellement dans le gestionnaire des périphériques windows.

La syntaxe est : COM numéro de port: vitesse , parité , bits de stop , protocole

La vitesse est à adapter à votre interface :

GENLI : 9600 bauds sans protocole (0)

LI100 : 9600 bauds

LI100F : 9600 ou 19200

LI101 / LZV200 : 19200 ou 38400 ou **57600 (par défaut)** ou 115200 bauds. Protocole matériel rts-cts (2)

La parité est toujours sans (N), 1 bit de stop

Il est possible d'indiquer une **temporisation d'envoi** entre deux octets transférés à l'interface. Elle peut être nulle. Pour les interfaces série sans protocole (0) comme le GENLI, il faut la positionner à une valeur de l'ordre de 30 (ms). Pour les interfaces avec protocole matériel RTS-CTS (2) cette variable est ignorée.

La variable suivante définit une valeur maximale par tranche de 100 ms qui définit la **temporisation d'attente** de la réponse de l'interface après une trame qui lui est transférée. Cette valeur est à tester en fonction de votre interface. En cas de dépassement de la valeur, un message « pas de réponse de l'interface » sera affiché. Exemple 7=700ms d'attente maxi.

La variable **entête** permet d'intercaler ou non des octets de synchronisation nécessaires aux différentes interfaces. En principe avec l'utilisation d'interfaces série (GENLI), cette valeur doit être à 0. Pour les interfaces utilisant nativement de l'USB, cette valeur doit être à 1. La valeur 2 est utilisée exclusivement pour des interfaces à base d'arduino pour xpressnet.

**Accès Ethernet à la centrale :** Uniquement pour les centrales utilisant XpressNet par ethernet IP ou WIFI ou compatibles. Ces champs contiennent l'adresse IP et le port de l'interface. L'adresse IP montrée ci-dessus (192.168.1.23) correspond à une interface Lenz\_USB-ETH qui a été configurée avec cette adresse évidemment. Mettre 0 si ou ne pas connecter l'interface ethernet si on ne veut pas l'utiliser.

**A propos des connexions par réseau Ethernet ou wifi aux centrales (uniquement pour le mode autonome):**

Il est préférable d'utiliser une connexion réseau, beaucoup plus rapide et plus fiable qu'une liaison série (RS232 ou USB). Si votre centrale ou votre interface vous permet de vous connecter par réseau, il faut privilégier ce mode de connexion.

**Initialisation des aiguillages :** En mode autonome, ce champ coché permet de lancer une séquence de positionnement des aiguillages au démarrage selon la description des champs "initialisation de l'aiguillage en mode autonome" de l'onglet aiguillages. Le séquençement entre deux aiguillages est donné par la variable "temporisation de séquençement d'init". La longueur de l'impulsion de pilotage de chaque aiguillage est donnée dans l'onglet "aiguillage" variable temporisation, car elle est spécifique à chaque aiguillage.

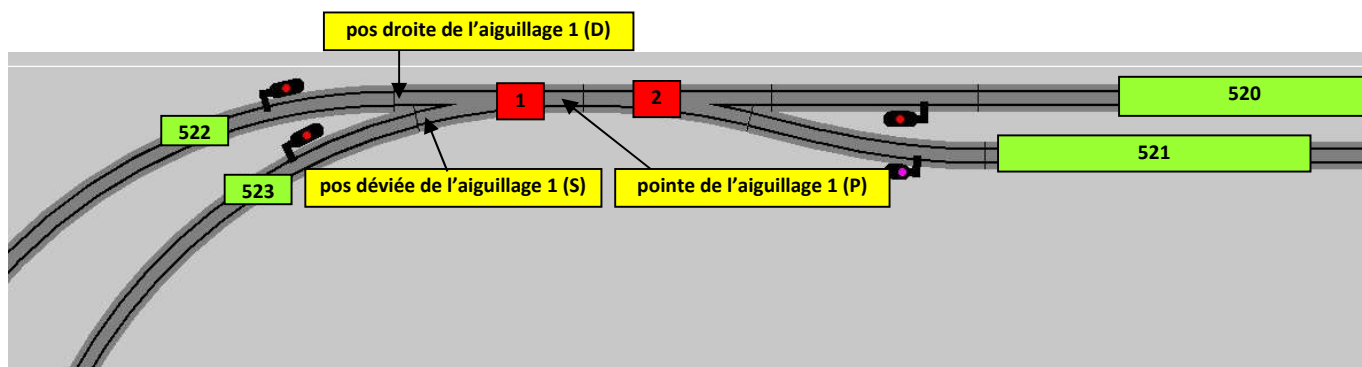
**Demande position des aiguillages à la centrale :** En mode autonome, ce champ coché permet au démarrage de demander à la centrale la position de chaque aiguillage, si la centrale en a connaissance. Cela permet de renseigner la position de chaque aiguillage dans signaux complexes. Il est conseillé d'utiliser l'option d'initialisation des aiguillages (ci-dessus). Dans ce cas, même si la demande de position des aiguillages est cochée, elle ne sera pas demandée.

**Connexion de l'interface Xpressnet en COM ou USB et Connexion de l'interface Xpressnet en Ethernet :** permet de connecter l'interface en COM ou USB ou Ethernet au démarrage de signaux\_complexes, si CDM n'a pas été lancé non plus au démarrage.

## Principes généraux de modélisation

Les aiguillages sont modélisés en décrivant les éléments connectés à leurs extrémités.

**Exemple :** soit l'extrait de réseau suivant, constitué de deux aiguillages d'adresses 1 et 2. Les zones de détection sont en vert et ont pour adresses 520, 521, 522 et 523. Peu importe la distance de l'aiguillage à la zone de détection (ou à un aiguillage), tant qu'elle est contigüe.



Définition des aiguillages : description des éléments connectés sur les 3 extrémités des aiguillages (P=pointe, D=droit, S=dévié). Pour créer un aiguillage, cliquer sur [nouveau], remplir l'adresse (Description de l'aiguillage). Choisir le type d'aiguillage (simple, TJD, TJS, Triple). Décrire les éléments connectés sur la pointe, la position déviée et la position droite. Exemple : la pointe de l'aiguillage 1 est "connectée" à la pointe de l'aiguillage 2, on renseigne 2P dans le champ P de l'aiguillage 1.

Les aiguillages 1 et 2 seront renseignés comme suit :

**Description de l'aiguillage 1**

Aiguillage simple [Restaurer]

Vitesse de franchissement dévié :   
☒ sans   
☐ 30 km/h   
☐ 60 km/h

Représentation :   
P [2P]   
S [523]   
D [522]

☐ Inversion de l'état CDM

Initialisation de l'aiguillage en mode autonome :   
Dévié ou droit : [Non positionné]   
Temporisation (1/10ème s) : [5]

**Description de l'aiguillage 2**

Aiguillage simple [Restaurer]

Vitesse de franchissement dévié :   
☐ sans   
☒ 30 km/h   
☐ 60 km/h

Représentation :   
P [1P]   
S [521]   
D [520]

☐ Inversion de l'état CDM

Initialisation de l'aiguillage en mode autonome :   
Dévié ou droit : [Dévié]   
Temporisation (1/10ème s) : [5]

La pointe de l'aiguillage 1 est connectée à la pointe de l'aiguillage 2

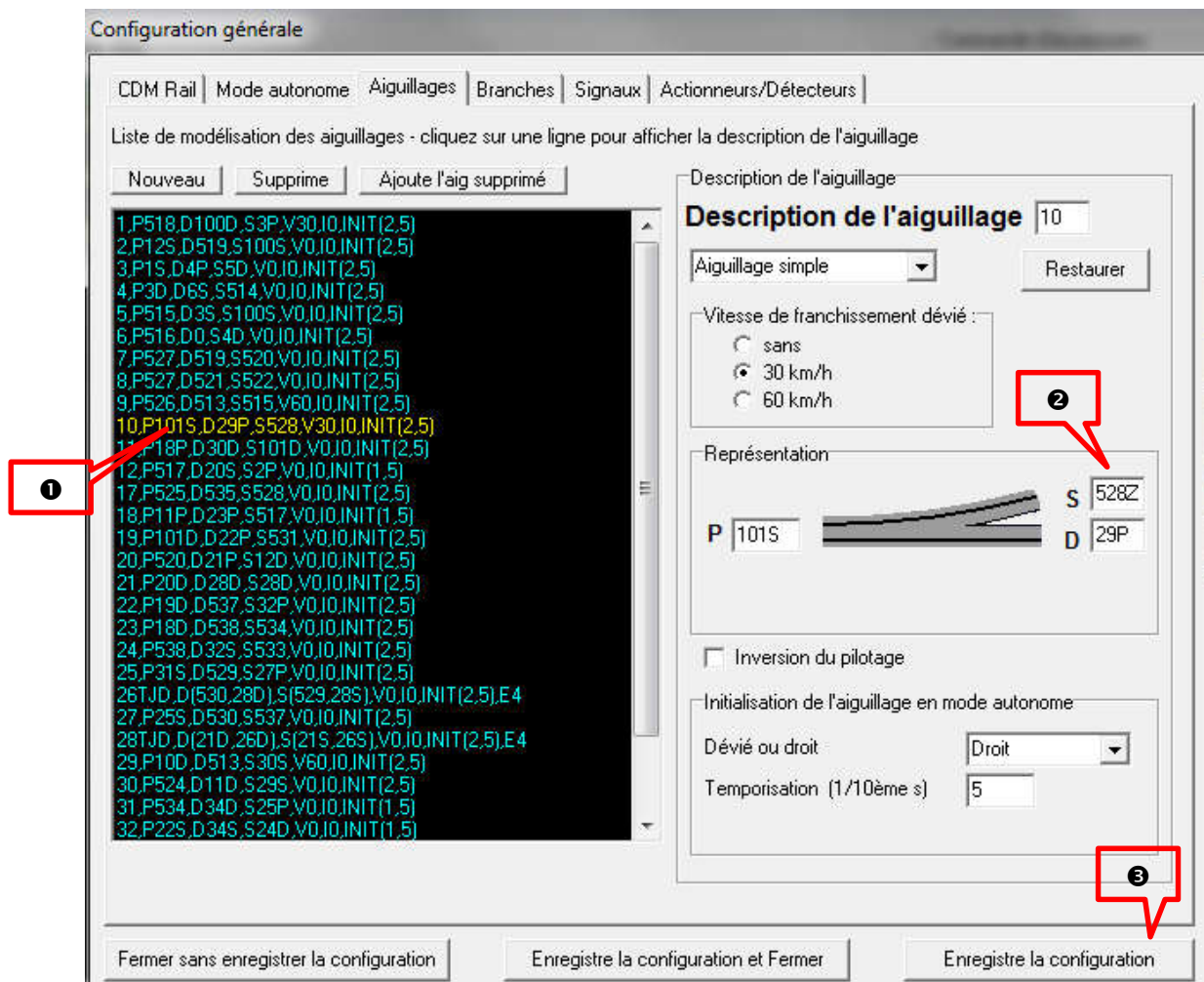
La position déviée de l'aiguillage 2 est connectée au détecteur 521

La position droite de l'aiguillage 2 est connectée au détecteur 520

2S désigne la position déviée de l'aiguillage 2 ; 2D désigne la position droite de l'aiguillage 2. 2P désigne la pointe de l'aiguillage 2. Voir la notice avancée pour d'autres exemples.



## Modification des aiguillages



Dans la liste, cliquer sur l'aiguillage à modifier ❶, l'élément apparaît en jaune.

Changer ensuite sa définition à droite ❷ (champs P pour pointe, S pour dévié, D pour droit, adresse, modèle, vitesse, inversion)

Dans la liste ❶, la couleur du texte change : Jaune = ligne modifiée – Blanc = ligne en cours (cliquée)

On peut changer la vitesse de franchissement de l'aiguillage en position déviée en cliquant sur l'un des 3 choix. On peut aussi changer son adresse.

Les modifications sur les aiguillages sont en mémoire. On peut changer d'aiguillage. En quittant la fenêtre, ne pas oublier de valider ❸ les modifications. Les fichiers de configuration seront écrits.

Certaines modifications des éléments nécessitent de vérifier la cohérence. Pour cela, fermer le panneau de configuration et sélectionner le menu "Divers / vérifier la cohérence", ou aller dans l'onglet branches et cliquer sur "vérification de la cohérence".

La modification de l'adresse d'un aiguillage, est conditionnée par le fait qu'une adresse soit libre.

Les détecteurs sont codifiés par leur adresse, ils peuvent également être suivis de la lettre Z (champ non obligatoire)

- On peut définir la vitesse de franchissement en position déviée (aucune, 30 ou 60 km/h, afin d'afficher le signal rappel 30 ou 60 lors de la présence d'un signal en amont de cet aiguillage). Pour que le train franchisse l'aiguillage en position déviée en vitesse réduite, il existe une option dans CDM Rail qui permet son paramétrage.
- Inversion du pilotage est à cocher si l'aiguillage est coché inversé dans CDM-rail (c'est-à-dire qu'il est câblé de façon inverse)
- La section "initialisation de l'aiguillage en mode autonome" sert à positionner (ou pas) l'aiguillage au démarrage de signaux\_complexes lorsqu'il est utilisé en mode autonome (sans CDM).

### Remarque importante

La position déviée (ou droite) décrite n'est pas forcément la position physiquement déviée (ou droite), par exemple si le câblage est inversé. Il faut que la position déviée corresponde à la commande nécessaire à l'obtention de la position déviée.

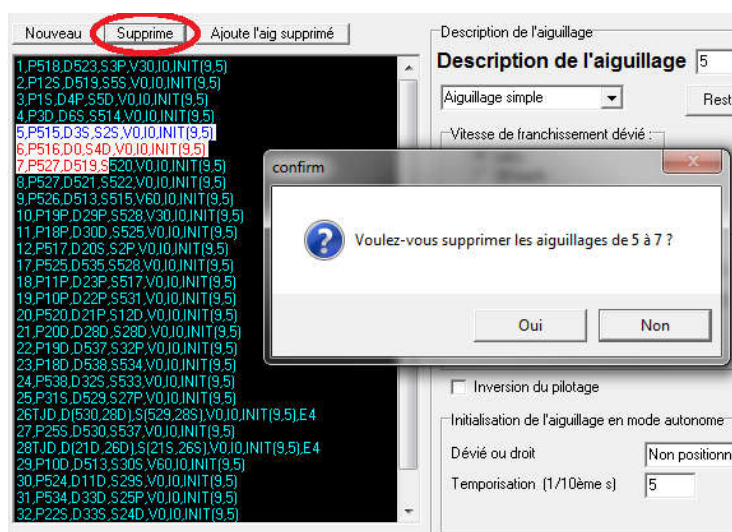
#### En câblage normal :

- + (2) correspond au pilotage de l'aiguillage en position droite (sortie 2 de l'adresse d'accessoire)
- (1) correspond au pilotage de l'aiguillage en position déviée (sortie 1 de l'adresse d'accessoire)

En principe, la coche « + » dévié dans CDM est **décochée**

Si le câblage est inversé, il faut inverser les positions dévié (S) et droite (D) dans la description.  
Pour éviter toute anomalie, il est vivement conseillé de mettre le câblage conforme.

### Suppression simultanée d'aiguillages

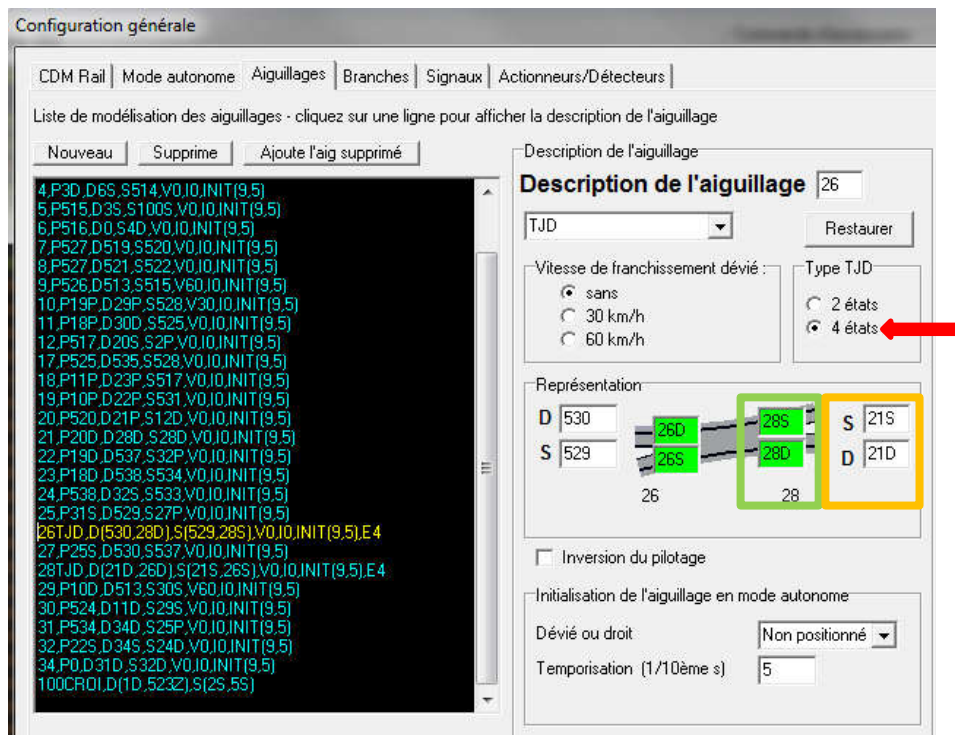


Il est possible de supprimer un groupe d'aiguillages. Pour cela sélectionner avec la souris la liste des aiguillages à supprimer (ici 5 à 7) en cliquant-tirant puis cliquer sur **Supprime** ou appuyer sur la touche Suppr.

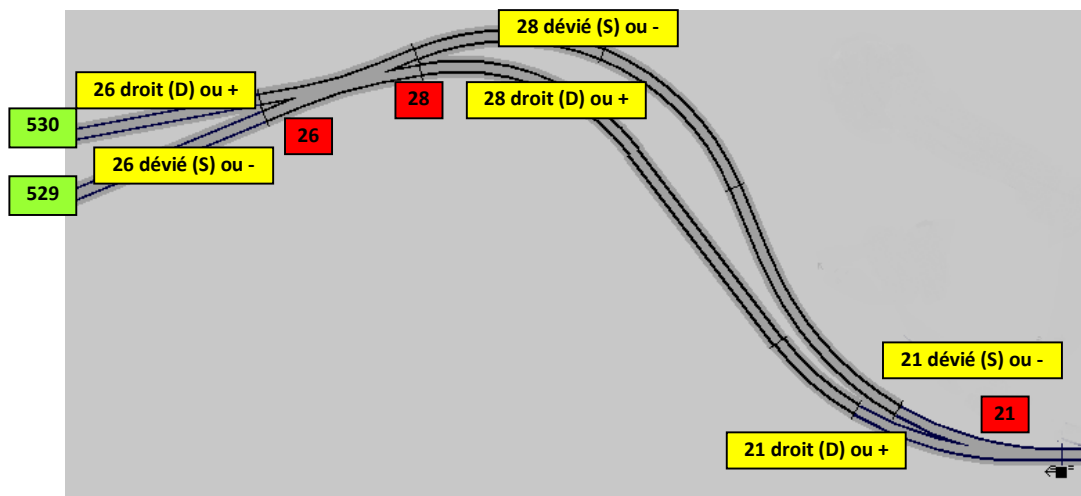
## Modélisation des TJD

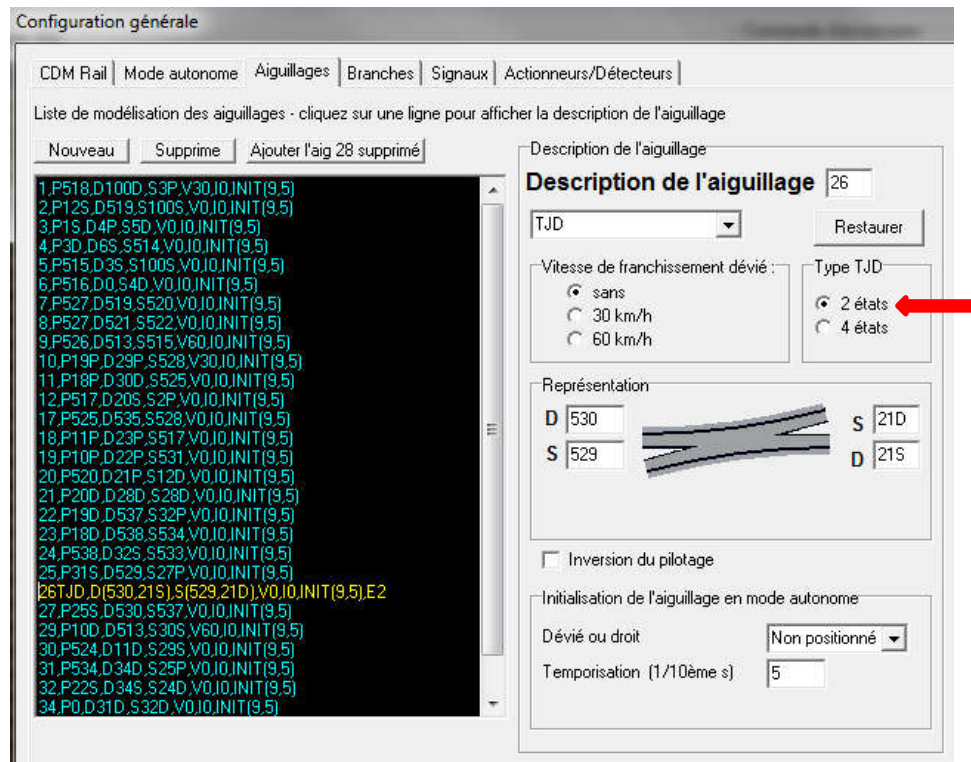
Les TJD 4 états occupent deux lignes car deux adresses sont affectées aux TJD. Exemple, ci-dessous la TJD 26 est définie par deux lignes, les aiguillages @26 et @28. La deuxième adresse de la TJD est définie dans l'encadré vert. Si on modifie l'un des champs de la zone verte, cela va modifier l'adresse de l'homologue de la TJD (l'autre ligne). Si cette adresse n'existe pas, la ligne sera créée. Si la création n'est pas désirée, il suffira de cliquer sur la ligne de la TJD créée et de cliquer sur [supprime].

Si on modifie l'un des champs de la zone orange, cela va modifier les paramètres correspondants de la TJD homologue (ici l'aiguillage 26).



Modélisation d'une TJD à 4 états, elle est déclarée aux adresses 26 et 28 et occupe donc 2 lignes.

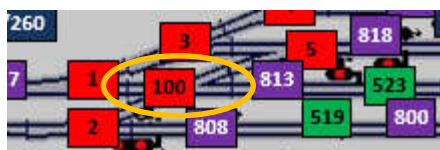
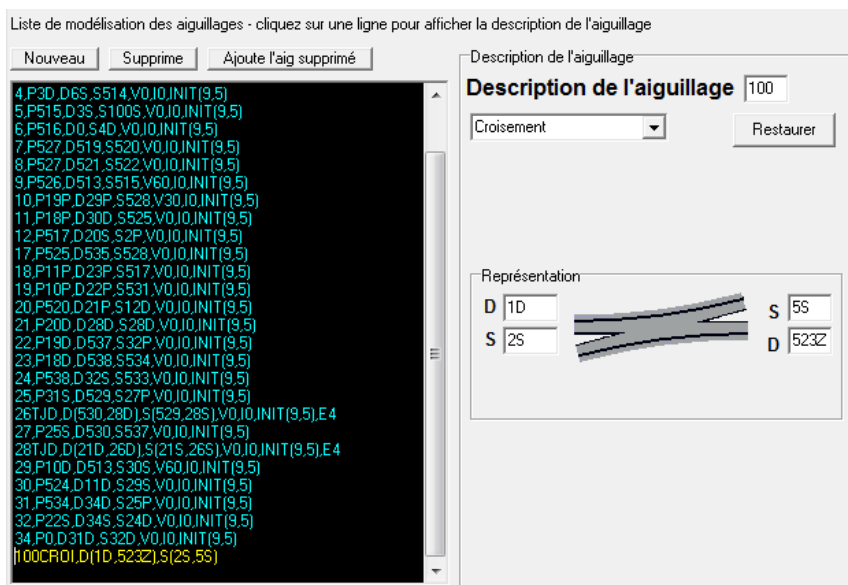




Une TJD à 2 états ne dispose que d'une seule adresse, et n'occupe donc qu'une ligne de déclaration.

## Modélisation des croisements

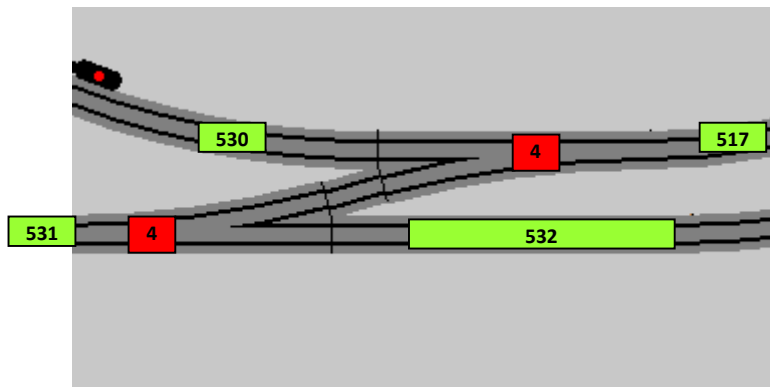
Les croisements sont apparus depuis la version 4.1. Ils permettent de gérer le croisement des trains pour en piloter les signaux contigus sur les voies concourantes au croisement. Bien qu'un croisement ne possède aucun moteur de pilotage, il est nécessaire de lui attribuer une adresse fictive unique.



Exemple de modélisation du croisement 100 entouré en orange

## Modélisation d'aiguillages ayant la même adresse

Cette description montre une utilisation non recommandée car CDM rail ne renvoie pas systématiquement la position des deux aiguillages lors de la manœuvre de l'un des deux aiguillages (bien qu'ils soient à la même adresse, ils bougent physiquement en même temps). La position des deux aiguillages n'est donc transmise de façon incorrecte au client. Cela entraîne des dysfonctionnements sur la localisation des trains. La bonne pratique est de mettre les aiguillages à deux adresses différentes. (Chaque aiguillage a son adresse). Un sujet est ouvert ici à ce sujet : <http://cdmrail.free.fr/ForumCDR/viewtopic.php?f=13&t=3958>



Les aiguillages ci-dessus ont la même adresse dans CDM (4). Cette situation n'est pas supportée dans signaux complexes. Il faut utiliser deux adresses différentes.



## Modélisation des aiguillages triples

Sous CDM-rail, un aiguillage triple est configuré de la façon suivante :

**CONFIG. AIGUILLAGE  
ADRESSE DIGITALE**

ADRESSE 31

"+" = Dévié ☐

ADRESSE 2 27

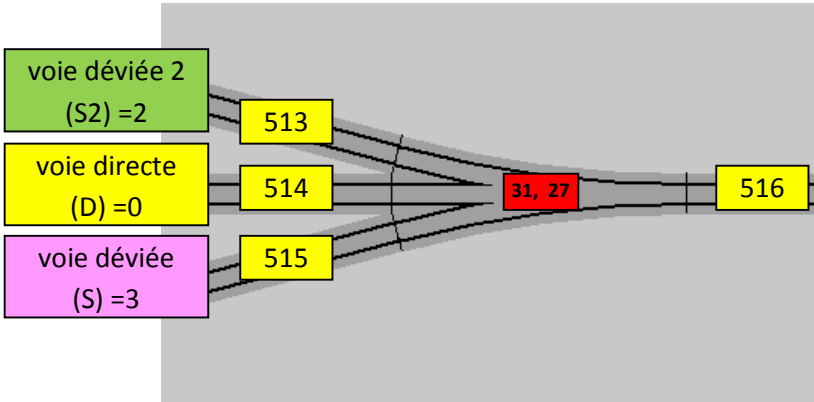
"+" = Dévié ☐

répéter commandes ☐

**DUREE IMPULSION**

Valeur/défaut: 0.1s

Modifier ☐



31 27	D	S
D	voie directe (0)	voie déviée (3)
S	voie déviée 2 (2)	-

D=droit      S=dévié

Dans l'exemple ci-dessus, l'aiguillage triple comporte les adresses 31 et 27. Cet aiguillage est « connecté » à des détecteurs (513, 514, 515 et 516)

adresse1 TRI, adresse2,  
 D élément connecté sur la voie directe,  
 S élément connecté sur la voie déviée (la voie gauche),  
 S2- élément connecté sur la voie déviée2 (la voie droite),  
 P élément connecté sur la pointe de l'aiguillage triple

Description de l'aiguillage

**Description de l'aiguillage** 31 27

31TRI,27,P516,D514,S515,S2-513,V30,I0

Aiguillage triple Restaurer

Vitesse de franchissement dévié :

☐ sans

☒ 30 km/h

☐ 60 km/h

Représentation:

P 516Z

S 515Z

D 514Z

S2 513Z

☐ Inversion de l'état CDM

Initialisation de l'aiguillage en mode autonome

Dévié ou droit 1

Temporisation (1/10ème s) 5

Description de l'aiguillage triple dans le panneau de configuration.

L'adresse de base est 31, l'adresse secondaire est 27.

Comme chaque aiguillage, on peut définir une vitesse limite de franchissement en position déviée.

Il ne doit bien sur pas y avoir d'aiguillage 27 préalablement déclaré.

Les éléments D, S, S2 et P peuvent être énumérés dans n'importe quel ordre. Par contre, l'ordre de déclaration de la 1<sup>ère</sup> adresse et de la 2<sup>ème</sup> adresse doit respecter le même ordre de déclaration que dans le tableau bleu de CDM-Rail. Ceci est automatiquement réalisé par le remplissage des éléments graphiques/

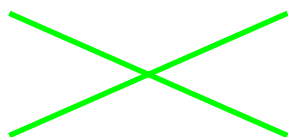
La voie droite d'un l'aiguillage triple est toujours la voie déviée 2  
La voie gauche d'un l'aiguillage triple est toujours la voie déviée « normale »



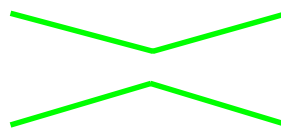
## Modélisation des TJD et des TJS

Il y a deux types de TJD: les TJD à 2 états (1 moteur) et les TJD à 4 états (2 moteurs)

### TJD2 états :



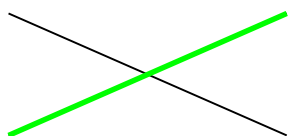
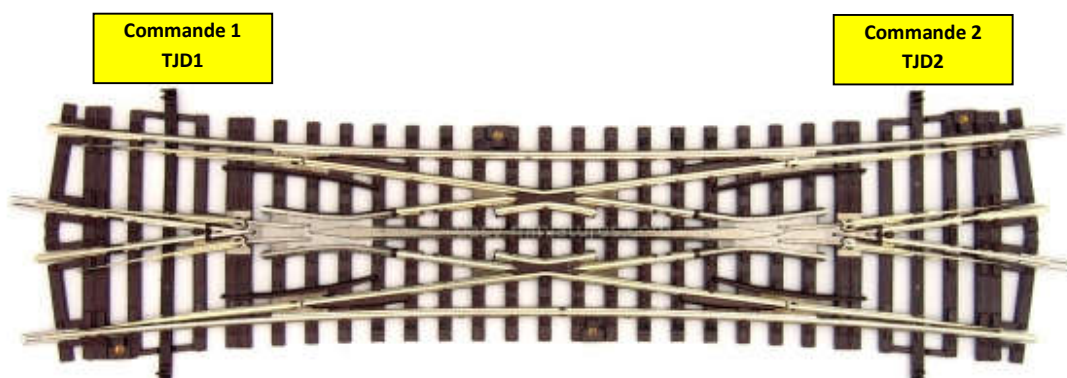
TJD droite



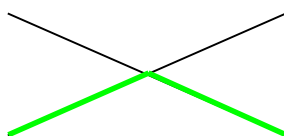
TJD déviée

### TJD 4 états :

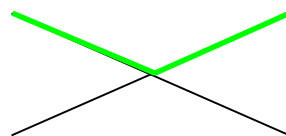
Elles comportent 2 moteurs pour 4 directions possibles indépendantes :



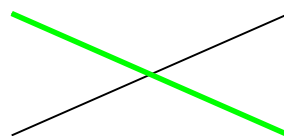
TJD1 déviée  
TJD2 déviée



TJD1 droite  
TJD2 déviée

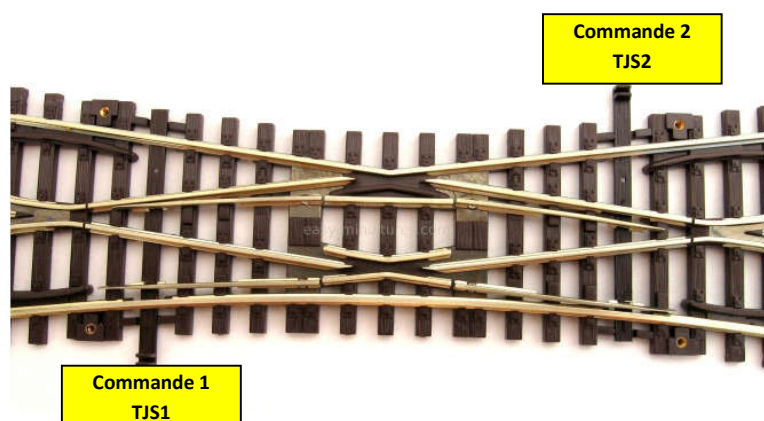


TJD1 déviée  
TJD2 droite

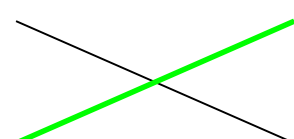


TJD1 droite  
TJD2 droite

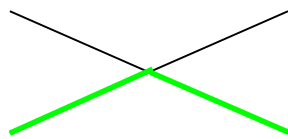
**Les TJS comportent 2 aiguilles pour 3 directions possibles :**



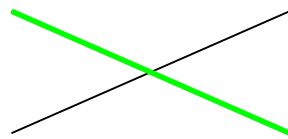
Directions possibles :



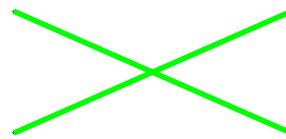
TJS1 déviée  
TJS2 déviée



TJS1 droite  
TJS2 déviée



TJS1 droit  
TJS2 droite



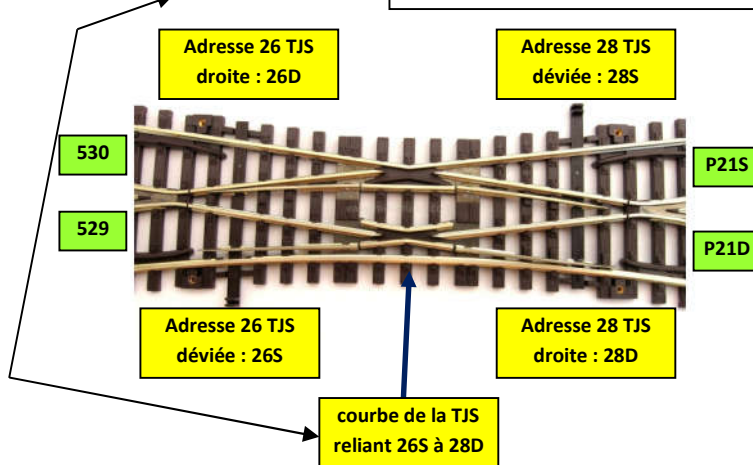
TJS1 déviée  
TJS2 droite

## Codification d'une TJS

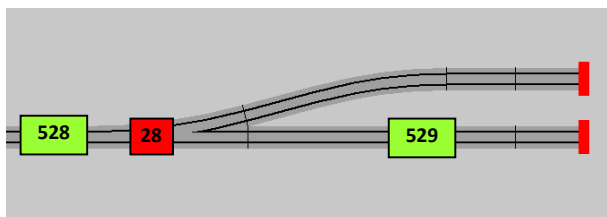
Une TJS comporte un paramètre supplémentaire L dans chaque ligne correspondant à l'endroit liant les éléments de la courbe de la TJS.

26TJD, D (530, 28D), S (529, 28S), L28D, V0, I0  
28TJD, D (21D, 26D), S (21S, 26S), L26S, V0, I0

éléments identifiant les éléments liés par la courbe de la TJS



## Modélisation d'un buttoir sur un aiguillage



Description de l'aiguillage

**Description de l'aiguillage** 28

Aiguillage simple

Vitesse de franchissement dévié :

☐ sans

☒ 30 km/h

☐ 60 km/h

Représentation

P 528 S 0 D 529

☐ Inversion de l'état CDM

Initialisation de l'aiguillage en mode autonome

Dévié ou droit

Temporisation (1/10ème s) 3

☐ Pilotage inversé

Le buttoir est connecté directement sur la branche déviée de l'aiguillage 28, on indique 0 dans le champ S.

Buttoir supérieur sans détecteur, buttoir inférieur avec détecteur. L'adresse 0 modélise un buttoir, connecté sur l'élément dévié (S) de l'aiguillage 28.

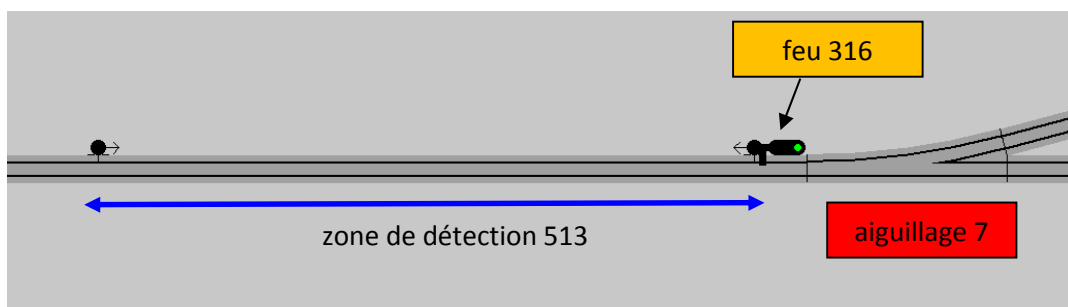
Dans les branches, le parcours 528 à 529 est modélisé comme suit :  
 . . . , 528 , A28 , 529 , 0    Le 0 terminal modélise un buttoir.

La description du parcours de 28S vers le buttoir n'est pas nécessaire.

L'élément 0 permet également de limiter le parcours à ce point. Par exemple, si la suite du parcours ne comporte plus de détecteurs ou d'aiguillages disposant d'une adresse (aiguillages strictement manuels).

## Modélisation des signaux

L'accès à la modification des signaux par le panneau de configuration se fait par le menu *divers/configuration*, onglet "signaux". Cliquer sur Nouveau pour ajouter un nouveau signal.



Le feu complexe d'adresse 316 à 9 feux est piloté par un décodeur digitalBahn est associé à la zone de détection 513. L'élément immédiatement suivant après le feu est l'aiguillage 7, et le feu est verrouillable au carré. Il sera modélisé comme suit :

La fenêtre 'Description du signal' pour le signal 316 contient les paramètres suivants :

- Description du signal** : 316
- Décodeur** : digital Bahn
- Aspect** : 9 feux
- Détecteur associé** : 1 513, 2, 3, 4
- Element suivant** : 1 A7, 2, 3, 4
- Restaurer** : bouton
- Verrouillable au carré** : ☒ (coché)
- Avec demande feu blanc** : ☐ (non coché)
- Conditions supplémentaires d'affichage du carré par les aiguillages** : zone de texte vide

Modélisation du feu 316 associé au détecteur 513, et suivi dans le sens de circulation du feu à l'élément suivant A7 (A pour aiguillage). Si l'élément suivant est un détecteur, on note uniquement son adresse (sans A)

Les lignes 1 à 4 concernent les voies communes à un signal. Lors que le signal ne concerne qu'une voie comme ici, on ne renseigne que la ligne 1.

**Nouveau** : Ajoute un nouveau signal à la fin de la liste à l'adresse arbitraire 999.

**Supprime** : supprime le signal cliqué droit après confirmation.

**Ajouter le feu supprimé** : Ajoute en fin de liste le dernier signal supprimé.

**Restaurer** : permet de restaurer l'état du feu en cours de modification au moment du clic dans la liste de gauche.

Pour utiliser un signal virtuel (donc non piloté), il faut sélectionner dans le champ "décodeur" la valeur "rien".

**Verrouillable au carré** : Ce champ ne concerne que les feux dont la forme est supérieure ou égale à 4 feux. Signal non verrouillable au carré (décoché) : le signal n'affichera pas de carré si aucun train n'est en approche sur les 3 cantons le précédent. Si le signal est verrouillable au carré (coché), le signal affichera un carré si aucun train n'est en approche sur les 3 cantons le précédent. **D'autre part, il est nécessaire de**

**décocher "verrouillable au carré" pour les signaux >5 feux qui n'affichent pas de carré (sémaphore uniquement).**

**Décodeur :** Contient le décodeur associé au signal. "Rien" désigne un signal virtuel.

**Avec demande feu blanc** permet d'afficher une case à cocher sur le signal "dem FB" (demande feu blanc). Dans ce cas, le feu vert sera remplacé par le feu blanc:



"avec demande feu blanc" n'est disponible que pour les feux d'aspect  $\geq 5$  feux.

La zone d'entrée "**conditions supplémentaires d'affichage du carré par les aiguillages**" contient des éléments permettant au feu d'afficher un carré si les aiguillages désignés ont la position indiquée. Exemple ci-dessous:

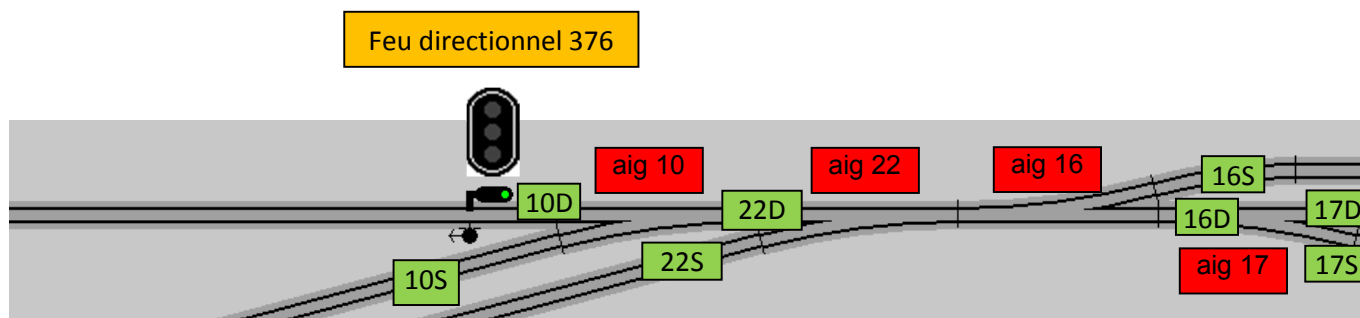
(6 lignes maximum)  
A21S,A6D  
A30S,A20D  
A1D,A2S,A3D  
(...)

(Aiguillage 21 dévié et aiguillage 6 droit) ou (aiguillage 30 dévié et aiguillage 20 droit) ou (aiguillage 1 droit et aiguillage 2 dévié et aiguillage 3 droit)

**Temporisation entre deux commandes :** permet de régler le temps envoyé entre deux commandes vers les signaux à CDM ou la centrale.

**Raz signaux après commande :** envoie un 0 au décodeur d'adresse après envoi d'une commande. Cette commande est normalement décochée.

## Signaux directionnels



Adresse du feu, D Nombre de directions, (Aiguillages mal positionnés en condition **ou**) (aiguillages direction la plus à gauche séparés par une virgule en condition **et**) (aiguillages direction 2 séparés par , en condition **et**) ... (aiguillages direction la plus à droite séparés par , en condition **et**)

Un feu directionnel à 3 feux aura 4 descriptions entre parenthèses. Ils ne sont pas liés à un détecteur.

Exemple pour un signal directionnel (D) à 3 directions d'adresse 376 piloté par un décodeur

Le feu à l'adresse 372 est un feu directionnel (D) piloté par un décodeur digitalBahn (1).

N'afficher aucun feu si l'aiguillage 10 est dévié (S) **ou** si l'aiguillage 22 est dévié (S) (car dans ce cas ils sont pris en talon et mal positionnés)

Afficher 1 feu si l'aiguillage 16 est dévié (S)

Afficher 2 feux si l'aiguillage 16 est droit (D) **et** si l'aiguillage 17 est droit (D)

Afficher 3 feux si l'aiguillage 16 est droit (D) **et** si l'aiguillage 17 est dévié (S)

Le nombre d'aiguillages décrits dans une parenthèse est illimité.

Description du signal

**Description du signal** 376

Décodeur: digital Bahn

Aspect: Directionnel 3 feux

Restaurer

Conditions d'affichage du feu directionnel :

A10S,A22S  
A16S  
A16D,A17D  
A16D,A17S

digitalBahn (1) :

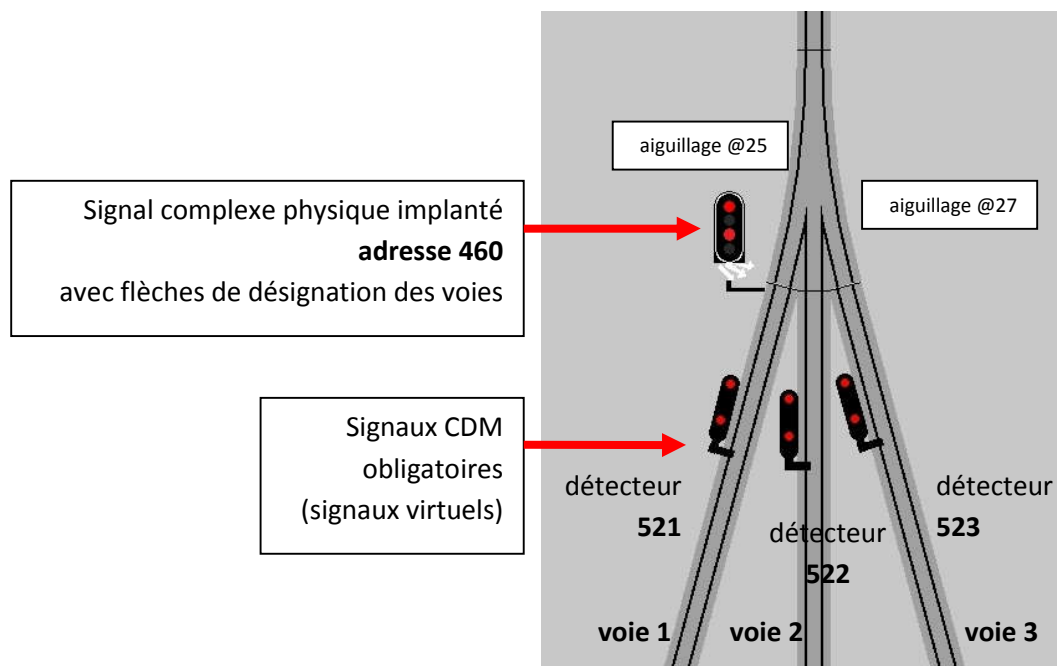
372,D3,1,(A10S,A22S)(A16S)(A16D,A17D)(A16D,A17S)

0 feu      1 feu      2 feux      3 feux

Attention la combinaison logique de l'extinction du feu est un **ou**.  
Les autres sont des **et**.

## Feux pour plusieurs voies simultanées

Cette configuration permet d'économiser des signaux sur des voies convergentes.



Le signal complexe indique la voie sur lequel l'aiguillage est positionné. Ce signal complexe doit comporter les 3 flèches de désignation des voies, signifiant qu'il implique les 3 voies. Cette configuration est utilisée pour économiser les signaux, notamment sur des faisceaux convergents en gare ou dépôts. Cette configuration est limitée à 4 voies maximum.

Description du signal

**Description du signal** 460

Décodeur: digital Bahn

Aspect: 7 feux

Détecteur associé	Element suivant
1 521	A25
2 522	A27
3 523	A27
4	

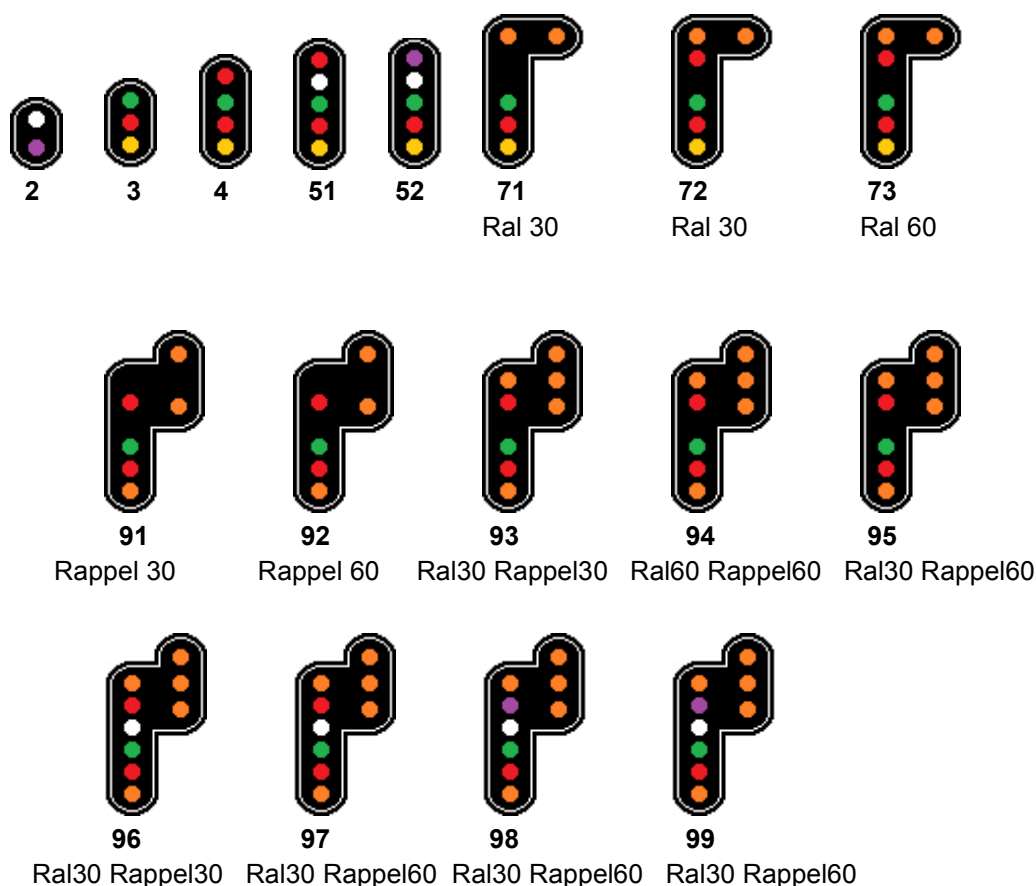
Restaurer

☒ Verrouillable au carré

Conditions supplémentaires d'affichage du carré par les aiguillages :

## Spécificités du décodeur Unisemaf

Ce décodeur nécessite un paramètre supplémentaire qui décrit précisément la cible. Il décrit la cible suivant le tableau ci-dessous (numéro inscrit sous la cible). De ce fait, le 2<sup>ème</sup> paramètre (ici 7) décrit uniquement le nombre de feux de la cible.



Description du signal

**Description du signal** 190

Décodeur: Unisemaf

Aspect: 7 feux

Détecteur associé	Element suivant
1 523	526
2	
3	
4	

Spec Unisemaf: 71

Ci-dessus, le décodeur unisemaf affecté au signal d'adresse 190 possède une cible "71" correspondant à l'affichage d'un ralentissement 30, sans affichage du carré. Une cohérence est faite entre l'aspect du signal et le paramètre Unisemaf.



### Information supplémentaire sur le décodeur unisemaf:

Ce décodeur se paramètre en modifiant ses variables de configuration (CV). Il faut brancher les signaux PQ (de la voie de programmation) sur le bornier X3.

#### Pour les possesseurs de centrale Lenz :

L'écriture des CV par le mode 8 de la raquette affiche ERR2 mais le CV est quand même écrit.

L'adresse du décodeur sur le bus DCC est stockée dans les CV1 et CV9 selon les règles et formules suivantes :

CV1 de 0 à 63 ; CV9 de 0 à 7

Dcc est l'adresse du décodeur sur le bus Dcc

$$CV1 = \frac{DCC-1}{4} + 1 \text{ modulo } 64 \qquad CV9 = \left( \frac{DCC-1}{4} + 1 \right) / 64$$

Modulo 64 est le reste de la division par 64

/ 64 est la division entière par 64

Inversement :

$$DCC = 4 \times (64 \times CV9 + CV1 - 1) + 1$$

$$DCC = 4 \cdot (64 \cdot CV9 + CV1 - 1) + 1$$

Exemple : Dcc = 281 CV1 = 7 et CV9 = 1

Il est possible d'affecter l'adresse du décodeur par apprentissage en appuyant sur son bouton, et ensuite envoyer à l'adresse désirée une commande d'accessoire avec le décodeur branché sur le bus Dcc. Attention l'adresse envoyée doit être modulo 4. La première adresse du décodeur ne peut prendre que l'une des adresses suivantes : 1 5 9 13 17 21 25 29 ... 2033 2037 2041. Si on envoie une adresse différente, l'adresse du décodeur sera ramenée à l'adresse modulo 4 inférieure.

Le changement d'aspect de ce décodeur est obtenu selon le tableau PACO – SNCF établi par Laurent Rieffel. (Doc Unisemaf648). Les CV à partir de 35 doivent être programmés selon ce tableau. Exemple pour afficher un carré sur un signal 51, il suffit d'envoyer une commande + (2) à l'adresse de base du décodeur +1 soit 52.

[http://usuaris.tinet.cat/fmco/download/UniSemaf648\\_manual.pdf](http://usuaris.tinet.cat/fmco/download/UniSemaf648_manual.pdf)

Il existe un paramètre dans le fichier de configuration (qui n'apparaît pas dans le panneau de configuration) qui permet de choisir l'algorithme de pilotage des décodeurs Unisemaf. C'est le paramètre Alg\_Unisemaf. Si ce paramètre vaut 1, l'algorithme non officiel mais testé et fonctionnel sera utilisé.

Si ce paramètre vaut 2, l'algorithme officiel de la documentation ci-dessus mais non testé sera utilisé.

Ce paramètre n'apparaît qu'à partir de la version 3.7.

## Spécificités du décodeur Stéphane Ravaut (SR)

Lorsque l'on sélectionne le décodeur SR dans la description du signal, un bouton "configuration" apparaît:

Il permet d'associer les fonctionnalités du signal par adresse occupée, par exemple :

Configuration du décodeur du signal Stéphane Ravaut

Adresse	Etat	Aspect	CV	
700	+ 2	<input type="text" value="carré"/>	CV=129	CV=129
	- 1	<input type="text" value="violet"/>	CV=1	CV=1
701	+ 2	<input type="text" value="sémaphore"/>	CV=136	CV=136
	- 1	<input type="text" value="sémaphore cli"/>	CV=136	CV=8
702	+ 2	<input type="text" value="jaune"/>	CV=72	CV=72
	- 1	<input type="text" value="jaune cli"/>	CV=72	CV=8
703	+ 2	<input type="text" value="vert"/>	CV=12	CV=12
	- 1	<input type="text" value="vert cli"/>	CV=12	CV=8
704	+ 2	<input type="text" value="blanc"/>	CV=2	CV=2
	- 1	<input type="text" value="blanc cli"/>	CV=2	CV=0
705	+ 2	<input type="text" value="ralen 30"/>	CV=32	CV=32
	- 1	<input type="text" value="rappel 30 + jaune"/>	CV=104	CV=104
706	+ 2	<input type="text" value="ralen 60"/>	CV=32	CV=0
	- 1	<input type="text" value="rappel 30 + jaune cli"/>	CV=88	CV=24
707	+ 2	<input type="text" value="Non commandé"/>	CV=0	CV=0
	- 1	<input type="text" value="Non commandé"/>	CV=0	CV=0

OK

Cette configuration doit correspondre au paramétrage réalisé dans le décodeur par l'intermédiaire de ses CV. Avec ce décodeur, un feu peut occuper 8 adresses au maximum, c'est-à-dire qu'un signal peut afficher 16 états différents à choisir. Si la table de représentation du décodeur diffère de la table de représentation déclarée dans signaux\_complexes, le signal n'affichera pas l'aspect désiré.

## Autres paramètres modifiables des signaux

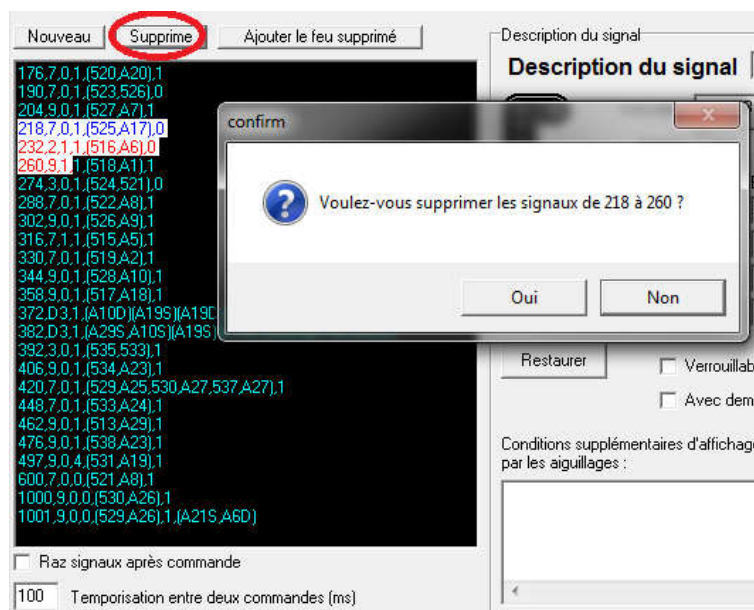
☐ Raz signaux après commande

100 Temporisation entre deux commandes (ms)

**RazSignaux** : Suivant la présence de la coche, envoie une commande 0 après l'écriture des décodeurs de signaux. Certains décodeurs autorisent de ne pas envoyer de commande à 0 après l'écriture dans le décodeur de l'aspect du signal, ce qui fait gagner du temps de pilotage des accessoires. Les décodeurs LEB nécessitent la remise à 0 après commande pour qu'ils soient pilotés correctement (ce qui est fait automatiquement dans signaux complexes, même si cette variable est décochée: Signaux\_complexes envoie systématiquement un 0 après pilotage d'une sortie d'un décodeur LEB. Ce champ devrait donc normalement être décoché.

**Temporisation entre deux commandes** : valeur de la temporisation en ms entre deux commandes envoyées au même décodeur.

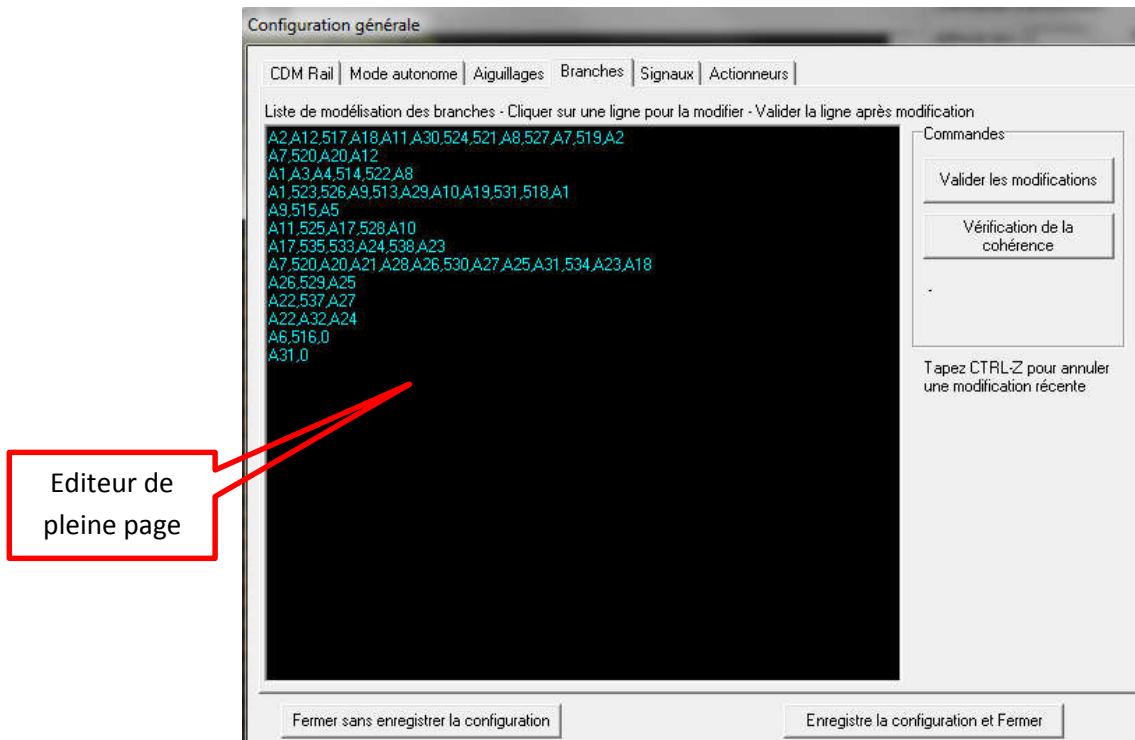
## Suppression simultanée de signaux



Il est possible de supprimer un groupe de signaux. Pour cela sélectionner avec la souris la liste des signaux à supprimer (ici 218 à 260) puis cliquer **Supprime** ou appuyer sur la touche Suppr.

## Modélisation des branches

Les branches de réseau peuvent être modifiées/créées depuis l'éditeur de textes de l'onglet "Branches" depuis le menu *Divers / configuration*.



Dans l'éditeur, on peut supprimer une ligne (soit une branche), le CTRL-Z (annulation) est fonctionnel à condition de ne pas cliquer ailleurs.

Il y a une branche de réseau par ligne.

Pour valider les lignes en mémoire après leur modification, cliquer sur "valider les modifications". Si les lignes ne comportent pas d'erreur, elles apparaîtront en vert, sinon les erreurs seront affichées en rouge.

Pour vérifier que la configuration générale est correcte (aiguillages, signaux et branches), cliquer sur vérification de la cohérence. En cas d'anomalie, elles seront affichées en rouge dans la fenêtre principale.

Une ligne constitue une branche du réseau.

On y décrit les détecteurs et les aiguillages. Les détecteurs sont renseignés par leur adresse. Les aiguillages sont renseignés par la lettre A suivi de l'adresse de l'aiguillage. Un buttoir est renseigné par 0.

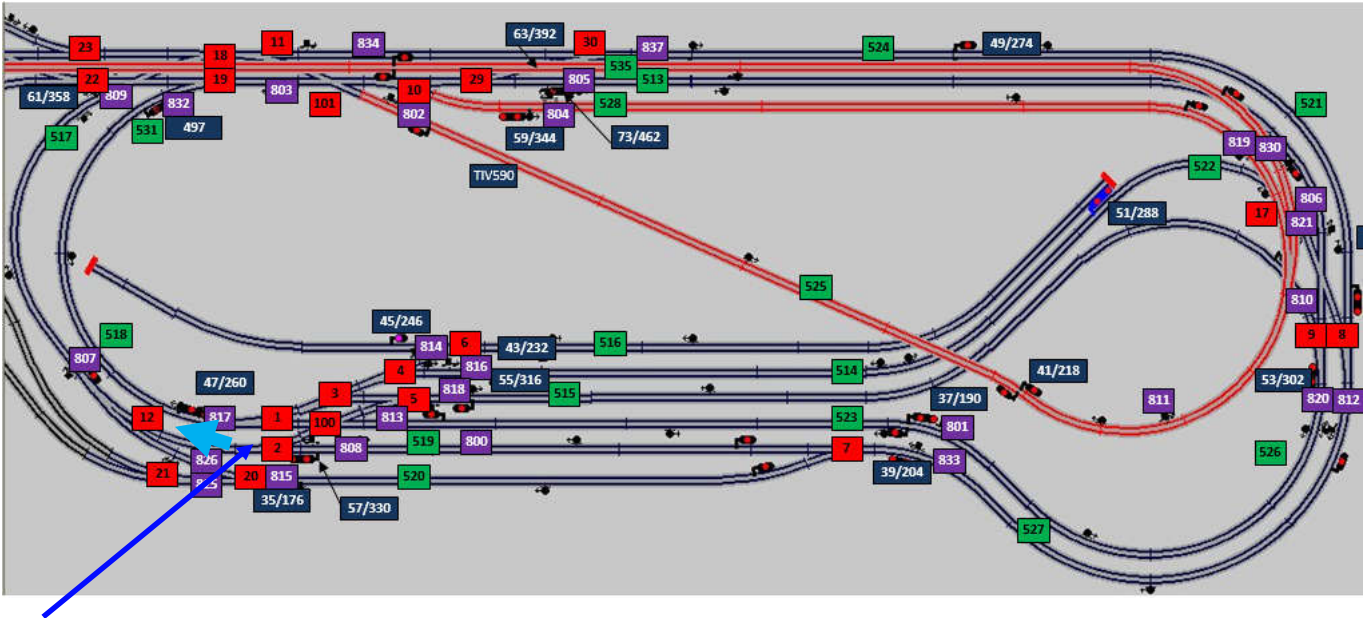
Le sens de description des branches est arbitraire.

Une ligne doit commencer par un aiguillage (ou un buttoir) et se terminer par un aiguillage (ou un buttoir). Il n'est pas nécessaire d'avoir un détecteur dans une ligne.

Un aiguillage peut se retrouver à plusieurs endroits de cette section, mais pas un détecteur. Tous les aiguillages déclarés doivent apparaître au moins une fois dans les branches.

Il y a bien sûr plusieurs façons de réaliser une description de branches, notamment sur l'endroit d'où on commence pour décrire les boucles de même que le sens de parcours.

Exemple de modélisation de ce réseau :



A2, A12, 517, A18, A11, A30, 524, 521, A8, 527, A7, 519, A2

A7, 520, A20, A12

A1, A3, A4, 514, 522, A8

A1, A100, 523, 526, A9, 513, A29, A10, A101, A19, 531, 518, A1

A9, 515, A5

A11, A101, 525, A17, 528, A10

A17, 535, 533, A24, 538, A23

A7, 520, A20, A21, A28, A26, 530, A27, A25, A31, 534, A23, A18

A26, 529, A25

A22, 537, A27

A22, A32, A24

A6, 516, 0

La première ligne est une boucle (de A2 à A2), elle décrit l'anneau extérieur du réseau. On part de l'aiguillage A2 désigné par une flèche. L'élément suivant est l'aiguillage 12 (A12). On ne décrit pas les pointes ou positions déviées ou droite de l'aiguillage car cela a déjà été fait dans la section de description des aiguillages

La deuxième ligne est une branche non bouclée. On part de l'aiguillage A7 en bas jusqu'à l'aiguillage A12 en bas à gauche.

La dernière ligne est un parcours vers le buttoir, elle est donc terminée par 0.

A100 et A101 sont des croisements.

## Modification évènements actionneurs/détecteurs

Cette section décrit les actions devant être déclenchés lorsqu'un train active ou désactive un actionneur ou un détecteur pendant le RUN de CDM rail. Il existe deux types d'actions possibles :

- Les actions déclenchées par un actionneur/détecteurs ou un détecteur pour *Fonctions F*, accessoires et sons.
- Les actions déclenchées par un actionneur/détecteur pour les *passages à niveaux* (PN).

### Différences entre actionneurs et détecteurs:

Les actionneurs ne sont disponibles que pendant un RUN de CDM rail et sont sensibles au sens de circulation. Le nom du train qui a déclenché l'actionneur est disponible.

Les détecteurs sont disponibles en mode autonome ou en RUN CDM rail et insensibles au sens de circulation. Le nom du train qui a déclenché le détecteur est indisponible.

En mode avec CDM, le déclenchement de l'action est provoqué par un actionneur, détecteur ou une zone contigüe à deux détecteurs, sur le nom de train déclaré dans les descriptions de convois de CDM. En mode autonome, seul un détecteur ou une zone contigüe à deux détecteurs peut déclencher l'action indépendamment de tout train.

Un actionneur est noté par son adresse. Un détecteur est noté par son adresse suivi de Z.

NB : Signaux complexes ne fait pas de différence entre le plan des adresses des actionneurs et des détecteurs. Un détecteur est en général dans la plage 513+, un actionneur dans la plage 800+ (ces derniers uniquement par convention).

On peut donc noter un détecteur simplement par son adresse sans qu'il soit suivi de Z.

Les évènements par actionneurs ou détecteurs ne nécessitent pas la modélisation du réseau. Dans ce cas on n'utilise pas les signaux, et les configurations d'aiguillage, de signal et de branches doivent être vides. C'est la configuration nulle.

Attention de ne pas disposer les actionneurs trop proches des zones de détection. En RUN, en cas de resynchronisation entre le train réel et le train de CDM rail en retard, un rattrapage est fait (resynchronisation) qui peut ne pas générer l'évènement d'actionneur. Il est toujours préférable que le train virtuel soit légèrement en avance sur le train réel pour éviter ce phénomène.

## Déclencher une fonction F d'une locomotive (F0 à F12 ou F28) , accessoire ou son

Description de l'actionneur

Type d'actionneur

☒ Fonction F pour locomotive  
☐ Accessoire  
☐ Son

Actionneur de fonction F de locomotive

Déclencheur

Type de déclenchement

☒ Actionneur/Détecteur  
☐ Zones de détection

Actionneur 815 à 1  
 DétecteurZ  
 Train Décl CC406526

Action

Action : Fonction 2  
 Tempo (ms) 200  
 Train Dest TGV

Test actionneur

### Actionneur de fonction F locomotive

#### Déclencheur

Actionneur/détecteur : adresse de l'actionneur ou du détecteur (suivi de Z) dans CDM Rail, celui-ci peut être "sensibilisé" à 0 ou à 1. Ou Adresse d'une zone contigüe de détecteurs.

#### Train déclencheur

Nom du train déclencheur de l'actionneur défini dans CDM rail ou X pour que la condition s'applique à tous les trains.

#### Action

Fx = numéro de fonction à activer dans le décodeur du train dont le nom a été défini dans le champ "Train Dest".

#### Train destinataire

nom du train destinataire qui reçoit la fonction F.

#### Tempo

Temporisation avant désactivation de la fonction Fx.

lorsque l'actionneur 815 est mis à 1 par le train CC406526, déclenchement de la fonction F2 dans le train TGV. La fonction F2 retombe après 200 ms.

Ce qui se traduit par l'envoi de la commande F2=1 vers le train TGV puis F2=0 200 ms plus tard. Si la valeur de la temporisation est 0, la fonction ne se désactive pas.

Description de l'actionneur

Type d'actionneur

☐ Fonction F pour locomotive  
☒ Accessoire  
☐ Son

Actionneur d'accessoire

Déclencheur

Type de déclenchement

☒ Actionneur/Détecteur  
☐ Zones de détection

Actionneur 815 à 1  
 DétecteurZ  
 Train CC406526

Action

Action : Accessoire 2 à 2  
☐ Remise à 0 après pilotage

Test actionneur

### Actionneur d'accessoire

#### Déclencheur

Actionneur / détecteur : adresse de l'actionneur ou du détecteur (suivi de Z) dans CDM Rail, celui-ci peut être "sensibilisé" à 0 ou à 1. Ou Adresse d'une zone contigüe de détecteurs.

#### Train :

Nom du train défini dans CDM rail ou X pour que la condition s'applique à tous les trains.

#### Action

Accessoire = adresse de l'accessoire à activer, et valeur de la sortie (1 ou 2 soit droit/dévié)

Lorsque l'actionneur 813 est mis à 1 par le train CC406526, la sortie d'accessoire 613 est mise à 2, puis remise à 0 si "remise à 0 après pilotage" est coché.

Ne pas utiliser pour piloter les aiguillages à bobines, car la sortie ne repasse pas à 0 si la coche n'est pas cochée (surchauffe et destruction de la bobine).

## Note concernant les fonction F:

En mode CDM rail, seules les fonctions de F0 à F12 sont utilisables. En mode autonome, les fonctions F0 à F28 sont utilisables.

Exemple de fonctions d'un décodeur de locomotive (répartition non standardisées)

F0=allumage des feux rouges/blancs  
F1=bruits moteurs1  
F2=klaxon aigu  
F3=klaxon grave  
F4=klaxon grave/aigu  
F5=KVB  
F6=éclairage  
F7=bruit moteurs2  
F8=éclairage cabine1  
F9=éclairage cabine2

F10=bruitage  
F11=bruitage attelage 1  
F12=bruitage attelage 2  
F13=bruit de purge d'air des freins  
F14=jingle 1  
F15=annonce en gare  
F16=jingle 2  
F17=annonce en gare  
F18=sifflet  
F20 à F28=réserve

## Actionneur/détecteur Son

Description de l'actionneur

Type d'actionneur

☐ Fonction F pour locomotive  
☐ Accessoire  
☒ Son

Actionneur d'accessoire

Déclencheur

Type de déclenchement



☒ Actionneur/Détecteur  
☐ Zones de détection

Actionneur  
DéclencheurZ 527Z à 1  
Train Décl ☒

Action

Nom du fichier son:

KLAXON\_3.WAV

Test actionneur

Les actionneurs ne sont utilisables qu'en mode RUN avec CDM Rail. Les détecteurs sont utilisables en mode autonome.

Joue le son « KLAXON\_3.WAV » si le détecteur 527 passe à 1, quelque soit le train déclencheur.

Le nom du fichier de son doit impérativement être en majuscules, et uniquement au format WAV. Si aucun chemin n'est spécifié, il doit être stocké dans le répertoire de signaux\_complexes.

Si le fichier n'est pas trouvé, un son d'erreur est joué.

Un bouton de commande permet de louer le son en local.

NB: si on utilise un détecteur, on ne peut pas définir de train déclencheur car CDM rail ne renvoie pas le nom du train qui l'a déclenché. Si on remplit le champ **Train Décl** alors qu'on a déclaré un détecteur, il n'en sera pas tenu compte.



## Déclencheur "zone de détection"

The screenshot shows a configuration window for a 'Déclencheur' (trigger). It is divided into three main sections:

- Type d'actionneur**: Three radio buttons are present: 'Fonction F pour locomotive', 'Accessoire', and 'Son'. The 'Son' option is selected.
- Actionneur d'accessoire**: This section contains a sub-section 'Déclencheur' with two radio buttons: 'Actionneur/Détecteur' and 'Zones de détection'. The 'Zones de détection' option is selected. Below this, there are input fields for 'Mémoire de Zone' with values '527Z' and '520', followed by 'à' and the value '1'. A 'Train' field contains the character 'X'.
- Action**: A label 'Nom du fichier son:' is above a text input field containing 'KLAXON\_2.WAV'. Below the input field is a play button icon.

Une zone de détection peut également servir de déclencheur.

Il faut dans ce cas spécifier deux détecteurs contigus. Ce déclencheur de zone sera activé lorsqu'une locomotive pénétrera dans la zone de détection de 527 vers 520, et dans le sens spécifié.

Cette fonctionnalité est valable en mode autonome ou en mode RUN avec CDM rail.

Lorsqu'on utilise les zones de détection, la condition s'applique à tous les trains.

### A propos des trains déclencheurs (mode Cdm Rail)

CDM rail ne renvoie le nom du train déclencheur que dans le cas d'un actionneur ou d'un détecteur qui passe à 1. Il est donc inutile de provoquer une action sur un actionneur qui passe à 0 avec un nom de train.

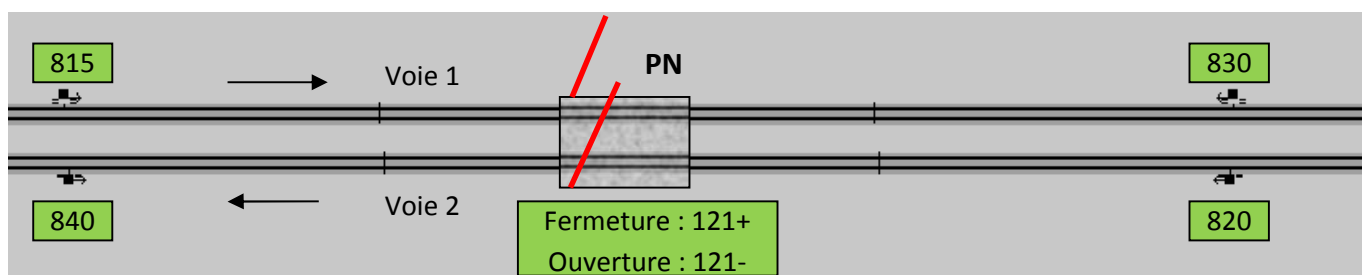
Pour les détecteurs, CDM ne renvoie pas systématiquement le nom du train sur un détecteur qui passe à 1. Provoquer une action sur un détecteur à 1 avec un nom de train sera aléatoire.

## Actionner un passage à niveau à une ou plusieurs voies

La fonction de commande des passages à niveaux est déclenchable soit par des actionneurs soit par des zones de détection. Le déclenchement de l'action « PN » est provoqué par n'importe quel train.

### Déclenchement PN par actionneurs, donc en RUN CDM rail

Le nombre d'adresses des actionneurs de fermeture et d'ouverture dépendent du nombre de voies que traversent le PN. Un passage à niveau à voie unique ne dispose que d'un seul détecteur d'ouverture et un seul détecteur de fermeture.



Actionneurs gestion passage à niveau

Adresse de fermeture 121 +

Adresse d'ouverture 121 -

	Act ferme	Act ouvre
Voie 1	815	830
Voie 2	820	840
Voie 3		
Voie 4		

	Zone ferme	Zone ouvre
Voie 1		
Voie 2		
Voie 3		
Voie 4		

Les actionneurs ne sont utilisables qu'en mode RUN avec CDM Rail. Les détecteurs sont utilisables en mode autonome.

Pour manoeuvrer un PN par les actionneurs, il faut renseigner les champs "Act ferme" et "Act ouvre".

La fermeture du PN est provoquée par la montée de l'actionneur 815 ou 820 et fait monter une mémoire de présence train sur chaque voie sur laquelle le détecteur 815 ou 820 a été activé.

La mémoire de présence train repassera à 0 après que les deux actionneurs 830 et/ou 840 se désactivent (queue du train).

Le sens de détection des actionneurs suit le sens de circulation de chaque voie.

Le passage d'un train sur la retombée de l'actionneur 830 et/ou 840 feront retomber la mémoire de présence train et le passage à niveau s'ouvrira (la zone des **deux** voies devra être dégagée).

La fermeture du PN est exécutée en envoyant + à l'adresse DCC 121.

L'ouverture du PN est exécutée en envoyant - à l'adresse DCC 121.

Dans le cas où les trains circulent dans les deux sens sur les deux voies, il faut ajouter dans CDM Rail sur les voies deux paires d'actionneurs supplémentaires par voie par sens, car les actionneurs sont sensibles au sens de passage du train, et ajouter leurs déclarations dans la ligne. On aura donc 4 actionneurs par voie.

L'ouverture du PN étant provoquée par la retombée du dernier actionneur de sortie (par la queue du train le franchissant), il n'est pas nécessaire de le mettre distant par rapport au PN.

Dans CDM, l'adresse de l'actionneur utilisée est son adresse (ici 809). Laisser cocher « out1 »

ADRESSE 809

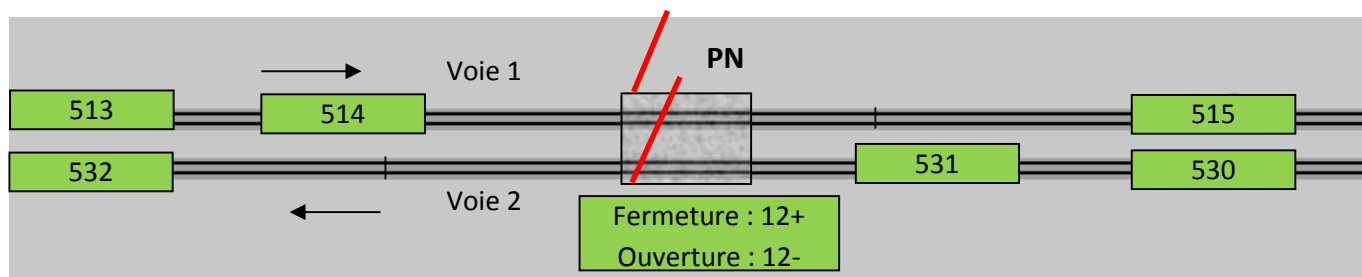
ACCESS. ACTION

ACC. STD. ☒

OUT1 ☒ OUT2 ☐

## Déclenchement PN par zones de détection, donc en mode autonome ou en RUN CDM rail

Le nombre des zones de détection de fermeture et d'ouverture dépendent du nombre de voies que traversent le PN. Un passage à niveau à voie unique ne dispose que d'une seule zone d'ouverture et d'une seule zone de fermeture. Une zone de détection est composée de deux détecteurs contigus (pour identifier le sens de déplacement du train).



Actionneurs gestion passage à niveau

Adresse de fermeture 12 +

Adresse d'ouverture 12 -

	Act ferme	Act ouvre
Voie 1	<input type="checkbox"/>	<input type="checkbox"/>
Voie 2	<input type="checkbox"/>	<input type="checkbox"/>
Voie 3	<input type="checkbox"/>	<input type="checkbox"/>
Voie 4	<input type="checkbox"/>	<input type="checkbox"/>

	Zone ferme		Zone ouvre	
Voie 1	513	514	514	515
Voie 2	530	531	531	532
Voie 3	0	0	0	0
Voie 4	0	0	0	0

Les actionneurs ne sont utilisables qu'en mode RUN avec CDM Rail. Les détecteurs sont utilisables en mode autonome.

Pour manœuvrer un PN par les zones de détection, il faut renseigner les champs "Zone ferme" et Zone ouvre".

**La fermeture du PN** est provoquée par l'activation des zones de détection de:

- 513 vers 514 lorsqu'un train circule de 513 à 514 (à la désactivation de 513)
- 530 vers 531 lorsqu'un train circule de 530 à 531 (à la désactivation de 530)

et fait monter une mémoire de présence train.

**L'ouverture du PN** est provoquée par la désactivation des zones de détection de:

- 514 vers 515 lorsqu'un train circule de 514 à 515 (à la désactivation de 515)
- 531 vers 532 lorsqu'un train circule de 531 à 532 (à la désactivation de 532)

La zone des **deux** voies devra être dégagée pour faire retomber la mémoire de présence de train.

La fermeture du PN est exécutée en envoyant + à l'adresse DCC 12.

L'ouverture du PN est exécutée en envoyant – à l'adresse DCC 12.

Le choix d'actionneurs ou de zones de détection dans le tableau ne peut être simultané. Le tableau "actionneurs" peut soit être rempli soit le tableau "zones" mais pas les deux.

## Erreurs à la lecture du fichier de configuration

Le fichier de configuration est lu au démarrage du programme signaux complexes. Certaines vérifications de cohérence sont faites lors de la lecture de ce fichier. Une erreur détectée dans celui-ci affichera un message d'erreur.

Les vérifications sont les suivantes :

Les lignes de la section des branches doivent commencer ou se terminer par un aiguillage ou un buttoir. Un aiguillage n'a pas décrit dans la section des branches du réseau mais il est présent dans la section description des aiguillages.

Un détecteur est décrit dans la description d'un aiguillage mais ce détecteur est absent dans la description des branches.

Etc..

## Erreurs à l'exécution

D'autre part des erreurs peuvent survenir pendant l'exécution du programme. La plupart du temps, elles proviennent d'une erreur de modélisation du réseau dans le fichier de configuration. Il est difficile de créer une configuration correcte dès le premier essai.

Néanmoins, des erreurs de fond du programme sont la deuxième cause d'erreur, qui ne sont pas liées au fichier. Ces erreurs, beaucoup plus rares, se produisent pour des cas particuliers. Ce sont :

131 - Erreur fatale : adresse nulle sur un aiguillage pris en pointe droit

134 - Erreur fatale : adresse nulle sur un aiguillage pris en pointe dévié

136 - Erreur fatale : adresse nulle sur un aiguillage pris en talon

139 - Erreur fatale - Aucun cas TJD/S : impossible de résoudre une TJD ou une TJS

Elément n non trouvé : un élément recherché (aiguillage ou détecteur) n'a pas été trouvé dans les branches

Erreur fatale X : Itération trop longue : une recherche sur un élément prend trop de boucles programme

X=300 201 200 201

Erreur X - feu non trouvé

X=600 650

601 - Signal « adresse » non renseigné

980 : Détecteur suivant introuvable - route x à y impossible à déterminer

981 : Détecteur précédent introuvable - route x à y impossible à déterminer

9999 : Erreur fatale, trop d'itérations

## Interfaces XpressNet

Les LI100 (Xbus) LI101F (xpressnet) et LI100F de Lenz sont des interfaces série



LI100 – LI100F



LI101F

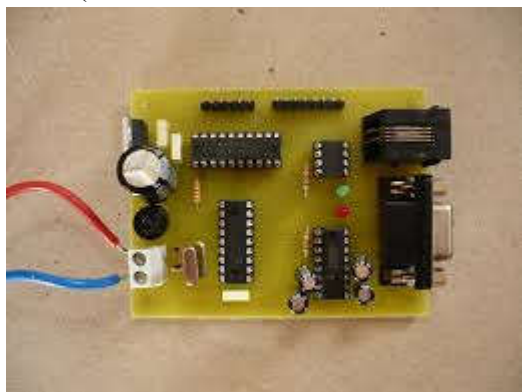


Interface USB (23150)



Interface USB – Ethernet (23151)

L'Interface Genli S88 est à la base une interface série, mais qui peut être dotée d'un convertisseur série-USB (mais elle reste à liaison série à la base)



## TCO (Tableau de commande optique)

Cette fenêtre permet de dessiner et d'exploiter un TCO. Un TCO d'exemple est fourni avec les fichiers du logiciel (TCO.CFG). Le TCO peut être ou pas affiché au démarrage du programme signaux\_complexes si l'option « avec TCO » est cochée dans le panneau de configuration (qui correspond à la variable « avec TCO » du fichier de configuration). Le fichier *TCO.CFG* est utilisé pour afficher le TCO. Un TCO comporte des éléments insérés dans des cellules cliquables ou sélectionnables par un tirer souris. Un rectangle de sélection bleu est alors affiché, et il peut être coupé, copié collé depuis le menu contextuel (clic droit).

Le TCO peut être configuré avec le bouton « configuration TCO ». On peut y définir les couleurs, la taille des cellules et leur nombre selon les valeurs limites suivantes :

	Taille des cellules		Nombre de cellules	
	Taille mini	Taille maxi	Nombre mini	Nombre Maxi
En X :	20	50	20	150
En Y :	20	50	10	70

La grille peut être ou non affichée.

Pour positionner des éléments dans le TCO, il faut faire glisser avec la souris en maintenant le clic gauche les éléments des icônes numérotées de 1 à 30 vers le TCO.

Les aiguillages, les signaux et les détecteurs peuvent être affectés à une adresse qui sont renseignés dans les champs en bas à gauche. Leurs textes peuvent être individualisés en couleur en cliquant sur "Fonte".

Les détecteurs sont uniquement portés par les éléments de voies 1, 6, 7, 8, 9, 10, 11, 16, 17, 18, 19 et 20. On peut avoir plusieurs éléments avec la même adresse de détecteur, et afficher le texte de l'adresse du détecteur ou non (choix dans la liste déroulante "position du texte"). Les styles de texte peuvent être changés en cliquant sur "Fonte" après avoir sélectionné une cellule.

Pour les signaux (élément 30), le fait de renseigner leur adresse dessine l'aspect du feu tel qu'il a été défini dans la configuration. Pour les faire tourner, il faut faire un clic droit sur le signal et sélectionner le menu rotation 90°G ou D ou vertical. On peut déclarer un signal à gauche ou à droite de la voie. Pour modifier la couleur de l'adresse du signal, il faut cliquer sur la cellule contenant l'adresse du signal (cas des signaux occupant deux cellules).

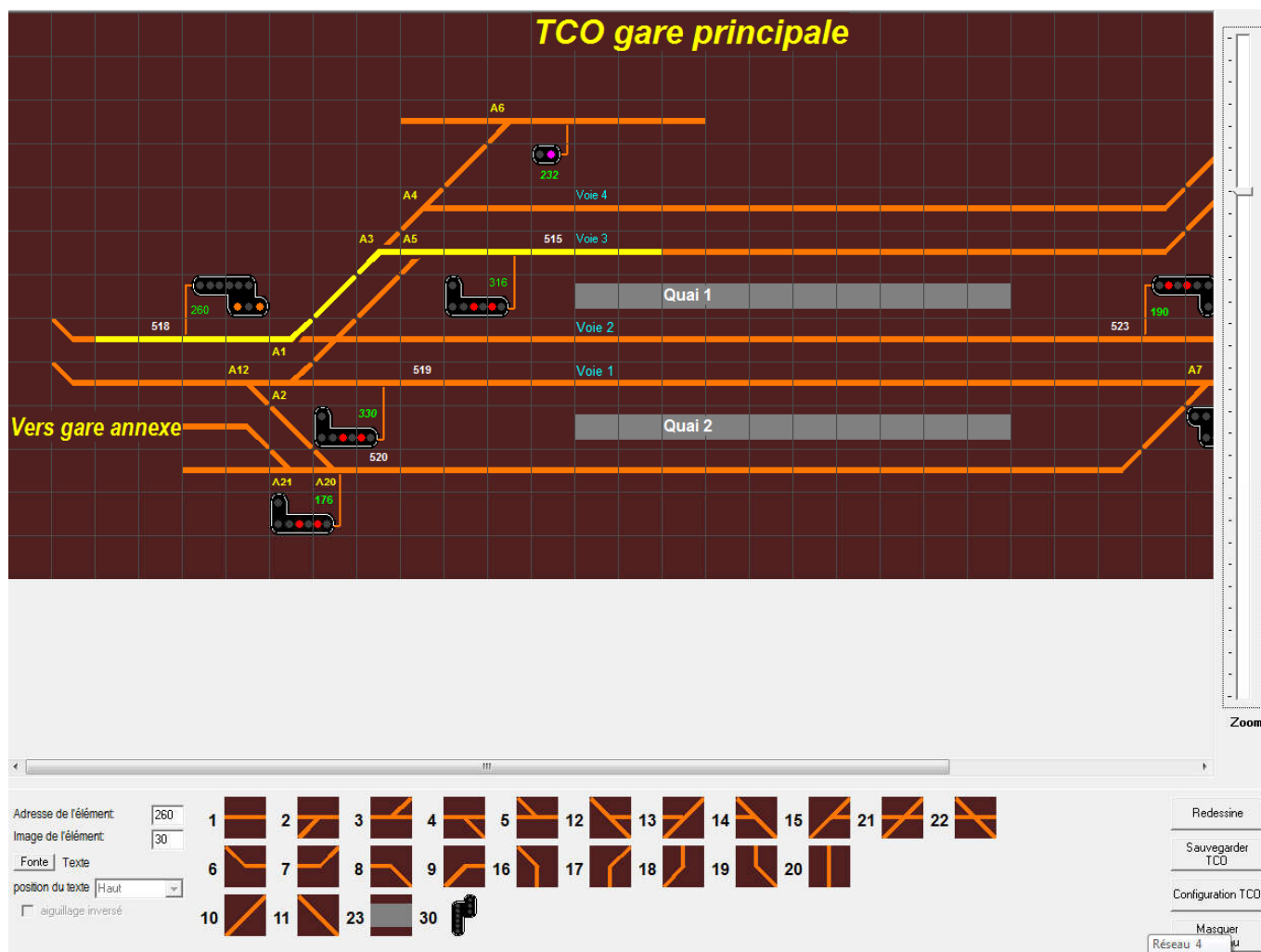
Des textes libres peuvent être affichés avec l'élément 1 ou dans toute autre cellule vide.

La barre de Zoom à droite est liée à la molette de la souris. La fenêtre TCO peut être redimensionnée à souhait.

En exploitation avec CDM rail en RUN ou en liaison directe avec la centrale via l'interface XpressNet, le TCO est dynamique. Il affiche :

- position connue des aiguillages,
- état des détecteurs et des zones entre deux détecteurs (« cantons »)
- feux des signaux.

On peut piloter un signal ou un aiguillage en double cliquant dessus.



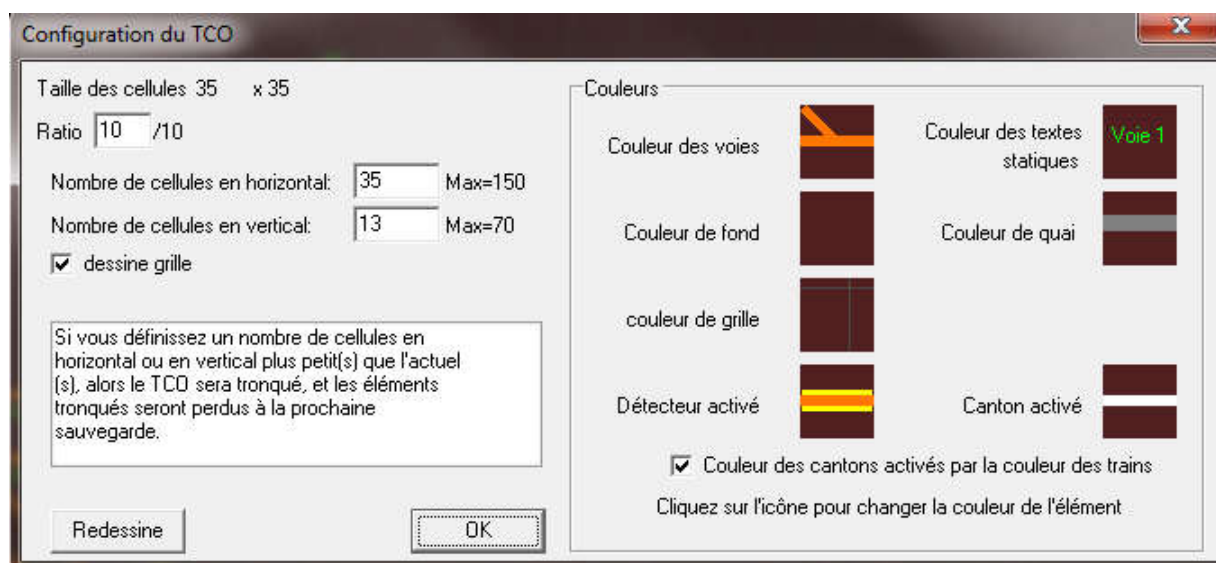
Vue d'un exemple d'écran de TCO,  
avec présence d'un train sur la zone de détection entre 518 et 515 (voie jaune)

Les éléments 21 et 22 sont soit des croisements ou des TJD-TJS. Pour les distinguer des croisements, il faut associer au composant l'adresse de base de la TJS-TJD dans le champ « adresse de l'élément ».

Pour les aiguillages (éléments 2 3 4 5 12 13 14 15) : un pilotage inversé doit être déclaré en cliquant sur la coche « aiguillage inversé » en bas à gauche.

L'élément 23 peut servir à représenter un quai.

Adresse de l'élément désigne l'adresse du détecteur, de l'aiguillage ou du feu associé à la cellule.



La sélection "couleur des cantons activés par la couleur des trains" permet d'afficher la couleur des cantons en fonction du numéro de train détecté, ou, si décochée, par la couleur du "canton activé" dans l'icône.

Le ratio est forcé à 10.



## Information sur les décodeurs d'accessoires en DCC

Le standard DCC définit le comportement des sorties sur les décodeurs d'accessoires, mais selon les fabricants, ce comportement est différent. En DCC, une adresse gère deux sorties, qui sont contradictoires, c'est-à-dire qu'elles peuvent être toutes les deux à 0, mais une seule peut être à 1 (et jamais les deux à 1). Chez Lenz, lorsque l'on commande une sortie à 1, l'autre sortie de la même adresse est mise à 0.

Exemple sur deux sorties d'accessoires à l'adresse 262 :

<b>262</b>	sortie 1 = 0	sortie 1 = 1	sortie 1 = 0	sortie 1 = 1
	sortie 2 = 0	sortie 2 = 0	sortie 2 = 1	sortie 2 = 1

Pour piloter la sortie 1 à 1, on demande la sortie 1 à 1 de l'adresse 262 (équivalent de l'appui de la touche – chez Lenz)  
Commande XpressNet : 52 41 8A 99

Configuration impossible  
en Lenz

Pour piloter la sortie 1 à 0, on demande la sortie 1 à 0 de l'adresse 262 (équivalent du relâchement de la touche – chez Lenz)  
Commande XpressNet : 52 41 82 91

Pour piloter la sortie 2 à 1, on demande la sortie 2 à 1 de l'adresse 262 (équivalent de l'appui de la touche + chez Lenz)  
Commande XpressNet : 52 41 8B 98

Pour piloter la sortie 2 à 0, on demande la sortie 2 à 0 de l'adresse 262 (équivalent du relâchement la touche + chez Lenz)  
Commande XpressNet : 52 41 83 90

52 est la commande de pilotage d'accessoire en XpressNet.

Les deux octets suivants (groupe et fonction) codifient la sortie, c'est à dire sa commande et son adresse.  
Le dernier octet est une somme de contrôle en ou exclusif.

sortie=0 ou 1

groupe:=(adresse-1) div 4;

fonction:=( ((adresse-1) mod 4)×2 + (sortie-1) ) or \$88 pour activer la sortie

fonction:=( ((adresse-1) mod 4)×2 + (sortie-1) ) or \$80 pour désactiver la sortie

La trame XpressNet est envoyée du PC à la centrale en USB ou Ethernet ou série, la centrale la transforme en trame DCC pour le bus JK, et le décodeur concerné réceptionne la trame DCC.



signal de ralentissement sur un réseau piloté par CDM Rail + programme client. Il annonce un aiguillage distant pris en pointe dévié à franchir à 30 km/h.  
Ce signal sera suivi d'un signal de rappel de ralentissement 30 km/h (deux feux jaunes verticaux) placé avant l'aiguille.



panneaux directionnels sur signaux complexes annonçant la direction que prendra le train.

## Utilisation du programme signaux\_complexes\_GL avec CDM rail

Les opérations suivantes sont automatiques si CDM rail est lancé depuis signaux complexes.

- Lancer **CDM rail**
- Lancer le serveur d'interface (**interface/démarrer un serveur**)
- Lancer le serveur IP (**comm IP/démarrer le serveur comm IP**).
- Lancer le programme client signaux\_complexes\_GL.

Il va alors afficher l'état des connexions :

### Exemple 1 :

Erreur 10065 socket IP: Port non connecté

Erreur 10061 socket IP CDM Rail: Connexion refusée

### Exemple 2 :

Erreur 10065 socket IP: Port non connecté

CDM Rail connecté avec l'ID XX

Le programme signaux\_complexes\_GL ne s'est pas connecté l'interface LENZ par réseau.

Le programme signaux\_complexes\_GL ne s'est pas connecté à CDM rail (CDM est pas lancé ou son serveur)

Le programme signaux\_complexes\_GL ne s'est pas connecté l'interface LENZ par réseau.

Le programme signaux\_complexes\_GL s'est connecté à CDM rail

Pour relancer une connexion à CDM rail, il faut le reconnecter en sélection le menu « interfaces / connecter CDM rail »

Dans CDM rail, passer en mode RUN avec ou sans trains (TCO).

Au lancement du mode TCO ou du RUN, CDM rail positionne les aiguillages. Ces positions sont alors récupérées par le programme client. (ainsi que chaque changement d'aiguillage par la suite).

Il faut un passage sur deux détecteurs consécutifs pour synchroniser un train avec le programme.

L'affichage dans le programme client donne l'état des pilotages des signaux et l'état des mémoires de zone. Il est possible de sélectionner l'affichage de l'un ou de l'autre dans le menu Debug et « afficher changement des détecteurs » (menus à bascule).

La fenêtre debug peut être fermée à tout instant et rappelée par le menu « affiche debug »

## Fenêtre DEBUG

La fenêtre debug permet d'afficher des informations sur le fonctionnement interne du programme. Il suffit de cocher une option dans la zone "sélection d'affichage" pour que les informations associées s'affichent pendant mode RUN. L'affichage peut vite être surchargé suivant les informations demandées.

Des fonctions primitives existent pour vérifier si la description du réseau est correcte.

**Etat signal suivant** : permet de connaître l'adresse et l'état du signal suivant au **signal** indiqué dans la case de droite.

**Etat canton suivant au signal occupé** : permet de connaître si le canton suivant au signal indiqué est occupé par un train.

The screenshot shows a window titled "Fonctions primitives". It contains four main sections, each with a label and a corresponding input field:

- Etat signal suivant**: An empty input field.
- Etat canton suivant signal occupé**: An empty input field.
- Détecteur suivant aux éléments**: An empty input field.
- Three numeric input fields containing the values **225**, **513**, and **519**.

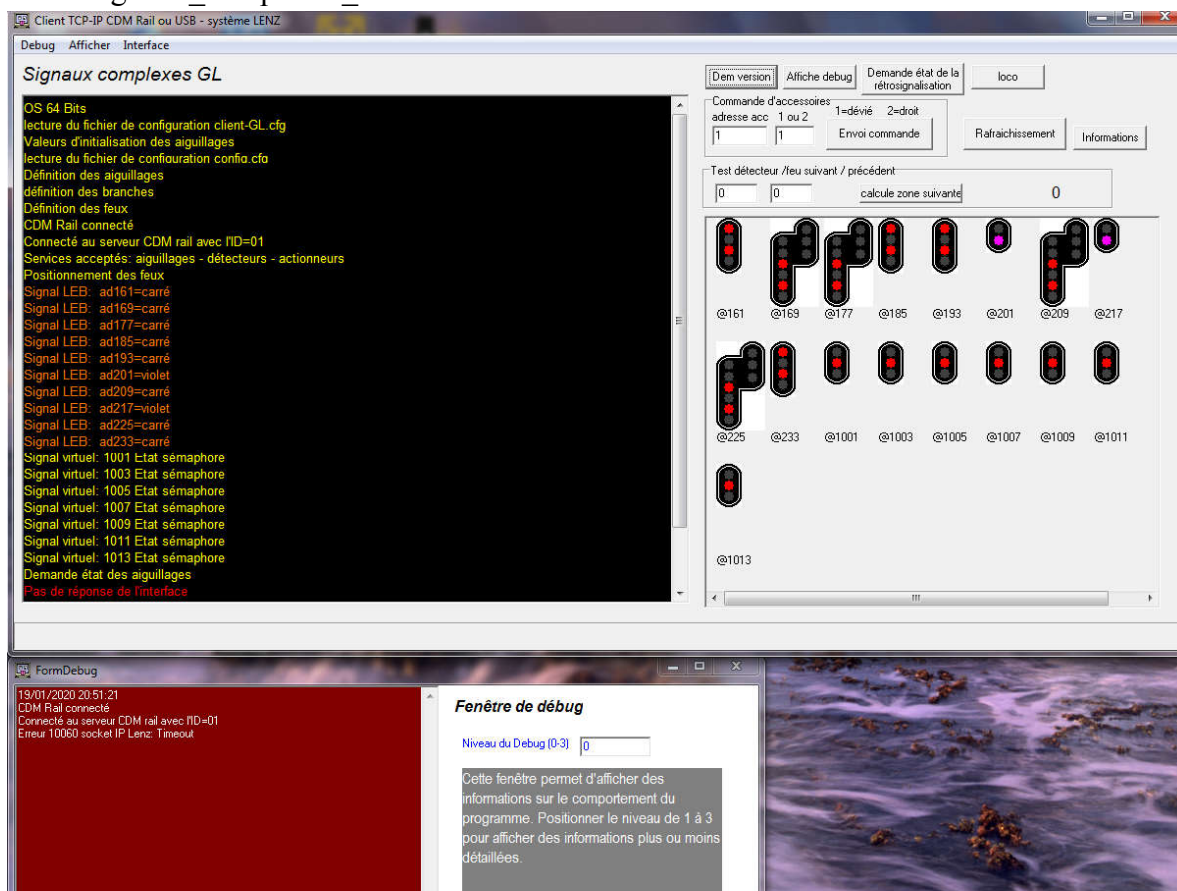
A red arrow points from the text "signal indiqué" in the paragraph above to the "225" field.

**Détecteur suivant aux éléments** indiqué dans les deux cases à droite : trouve le détecteur suivant aux deux éléments indiqués, dans le sens indiqué. Ci-dessus, on cherche le détecteur suivant aux détecteurs de 513 à 519. Les éléments peuvent ne pas être contigus. Un élément est soit un détecteur ou un aiguillage. Pour un aiguillage sa notation est son adresse précédée de A. Exemple A20.

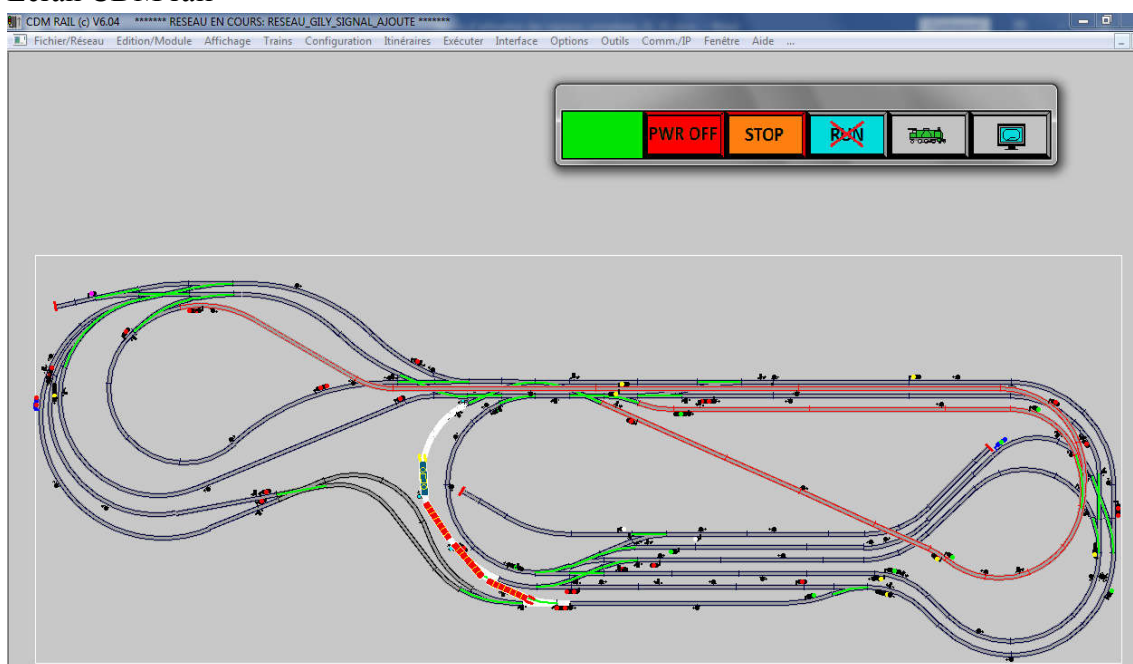
Cette fenêtre de fonctions primitives ne fonctionne que si la position des aiguillages est connue (début de RUN). Une erreur qui surviendrait pendant l'affichage de la fenêtre debug indique probablement une erreur dans la description.

Utiliser CDM rail et le programme signaux\_complexes\_GL peut occuper 3 fenêtres. Il sera donc opportun d'utiliser deux écrans de travail et de répartir les fenêtres sur les deux écrans.

## Ecran signaux\_complexes\_GL



## Ecran CDM rail



## Pilotage individuel des signaux

En cliquant sur l'image d'un signal dans la partie droite de la fenêtre, on ouvre la fenêtre de pilotage, et on peut sélectionner un aspect pour le signal choisi et envoyer la commande.



Les exclusions sont automatiquement gérées et ne seront pas affichées même si elles sont sélectionnées, comme par exemple ci-dessus : affichage d'un ralentissement 30 avec un avertissement n'est réglementairement pas possible.

## Vérification du pilotage des décodeurs

Il est possible de vérifier les séquences envoyées au décodeur de feu en fonction du feu (car un feu est associé à un type de décodeur). Pour cela ouvrir la fenêtre *debug* et cocher « affichage du pilotage des décodeurs de signaux » Lorsqu'un signal est piloté, sa séquence sera affichée dans la fenêtre debug :

**Fenêtre de debug** Niveau du Debug (0-3)

- ☐ Affichage du fonctionnement des signaux
- ☐ Affichage des événements détecteurs et aiguillages
- ☐ Affichage des événements actionneurs
- ☐ Affichage des évaluations des routes
- ☐ Affichage des trames échangées avec l'interface ou CDM
- ☐ Affichage des fronts descendants des détecteurs
- ☒ Affichage du pilotage des décodeurs de signaux

Tick=1001 signal 497 2  
 Tick=1001 signal 497 0  
 Tick=1001 signal 498 2  
 Tick=1001 signal 498 0  
 Tick=1001 signal 499 2  
 Tick=1001 signal 499 0  
 Tick=1001 signal 500 1  
 Tick=1001 signal 500 0  
 Tick=1001 signal 501 2  
 Tick=1001 signal 501 0

A gauche, séquence envoyée à un décodeur LEB à l'adresse de base 497, pour la commande de l'avertissement.

A droite, séquence envoyée à un décodeur Unisemaf pour une cible de type 52 à l'adresse de base 619, pour la commande de la voie libre.

Tick=4221 signal 619 1  
 Tick=4221 signal 619 0

Dans les deux cas, l'option « Raz signaux après commande » a été cochée.

## Pilotage des vitesses et des fonctions F des trains

On peut changer la vitesse des trains en **RUN** sur CDM rail ou en mode autonome. En mode CDM rail : choisir le train dans la liste déroulante et le % de vitesse et cliquer sur **envoi vitesse à loco**.

On peut piloter les fonctions F du décodeur d'une locomotive en indiquant son numéro de fonction de F0 à F12 (mode CDM) ou jusque F28 (mode autonome) et cliquer sur **envoi fonction à loco**.

En mode autonome, les noms des trains étant inexistant, il faut renseigner l'adresse du train.

## Simulateur

Le simulateur permet de simuler des circulations préalablement enregistrées. Dans cette étape, Il n'est pas nécessaire d'être connecté à CDM rail ou à la centrale via XpressNet.

### Il faut d'abord procéder à un enregistrement sur le réseau réel sur un parcours :

Faire circuler un ou plusieurs trains (en mode CDM ou sans CDM avec XpressNet). A la fin du parcours, aller dans la fenêtre debug dans le menu Divers « *Affiche fenêtre debug* » puis cliquer sur le bouton « *Affichage EvtS détecteurs et aig* ». Une liste d'évènements détecteurs et aiguillages s'affiche dans la fenêtre de gauche, correspondant au parcours réalisé. Cliquer sur le bouton « *Sauvegarder le log* » Ce bouton sauvegarde le contenu de la fenêtre de gauche ; donner un nom de fichier.

**Dans un deuxième temps**, à n'importe quel autre moment, on peut resimuler les parcours des trains dans signaux complexes\_GL depuis le fichier sauvegardé préalablement (les signaux seront pilotés).

Dans la fenêtre principale, sélectionner le menu *Divers / ouvrir un fichier de simulation*. Une fenêtre apparaît. Il est possible de choisir l'intervalle de temps entre deux détecteurs (temps comprimé).

Cliquer sur le bouton « *charger un fichier de simulation* » et sélectionner le fichier précédemment sauvegardé. La simulation commence dans 8 secondes.



## **Lecture / Ecriture de variables de configuration (CV)**

Ne fonctionne qu'avec des centrales dotées de bus **XpressNet version minimale 3.6** ou compatibles.

Il est possible de lire ou d'écrire des variables CV ou un fichier contenant des CV à un accessoire. Pour cela, le décodeur (train, feu) doit être branché sur la voie de programmation via les lignes PQ. Le programme doit être connecté directement à la centrale par USB ou Ethernet. CDM ne doit pas fonctionner (Signaux complexes est en mode autonome). On doit arrêter CDM rail et sélectionner le menu Interface/connecter USB ou interface/Connecter Ethernet pour se connecter à la centrale (pourvu que le fichier de configuration soit correctement renseigné : port COM usb ou adresse ethernet de l'interface XpressNet)

L'écriture des CV concerne les adresses de CV de 1 à 255. Pour les adresses >255, il faut utiliser XpressNet V3.6 mais ce n'est pas prévu dans le programme.

Lorsque la centrale est connectée par XpressNet via l'USB ou ethernet au programme (donc sans CDM), le boutons « Ecriture CV par le bus XpressNet » « Ecriture CV par le bus XpressNet » et le menu « Lire un fichier de CV vers un accessoire » sont utilisables.

### **Ecriture d'un CV seul dans un accessoire**

Indiquer l'adresse du CV dans le champ Adresse acc et la valeur du CV dans le champ 1 ou 2 et cliquer sur le bouton Ecriture CV. La centrale va passer en mode programmation (service mode) et écrire le CV. Pour repasser en mode DCC traditionnel, cliquer sur le bouton « Reprise DCC ».

### **Lecture d'un CV seul depuis un accessoire**

Indiquer l'adresse du CV dans le champ Adresse acc et cliquer sur le bouton Lecture CV. La centrale va passer en mode programmation (service mode) et lire le CV. La valeur du CV est affichée dans la fenêtre principale

Pour repasser en mode DCC traditionnel, cliquer sur le bouton « Reprise DCC ».

### **Ecriture de CV en cascade dans un accessoire depuis un fichier**

Un catalogue de CV peut être défini pour un accessoire dans un fichier texte en ASCII. Cela évite les manipulations fastidieuses si de nombreux CV doivent être écrits dans un accessoire, et permet de stocker les valeurs dans un fichier pour archivage. Cliquer sur le menu « Lire un fichier de CV vers un accessoire » et choisir le fichier. La centrale va passer en mode programmation (service mode) et écrire les CVs dans l'accessoire branché sur la voie de programmation. Pour repasser en mode DCC traditionnel, cliquer sur le bouton « Reprise DCC ». Attention l'envoi du fichier vers l'accessoire peut être très longue.

La structure du fichier doit contenir 2 champs numériques spécifiant l'adresse du CV suivie de sa valeur. Les champs non numériques sont ignorés, permettant des commentaires. Exemple :

Cv	valeur
-----aspect1	
35	63
36	36
37	0
38	0
-----aspect2	
39	63
40	33
41	0
42	0

## Lire un fichier de trames CDM

Cette fonction permet de lire un fichier trames issu de CDM rail au protocole Comm/IP. Il faut d'abord intercepter les trames de CDM rail pendant un RUN ou une circulation libre vers un fichier.

### Création du fichier trames

Pour cela, dans la fenêtre debug, cocher "Trames échangées avec l'interface ou CDM".

Lancer CDM rail et le serveur COMM IP.

Faire tourner un train en circulation, et changer les aiguillages. Tous les événements aiguillages, détecteurs et actionneurs échangés apparaîtront dans la fenêtre debug sous forme de protocole COM-IP.

Exemple:

```
Socket CDM rail connecté
C-C-00-0001-CMDGEN-__CNCT|000|
S-R-01-0001-CMDGEN-__ACK|000|
Connecté au serveur CDM rail avec l'ID=01
C-C-01-0002-RQSERV-RTSIM|030|03|SRV=ATNT;SRV=AACT;SRV=ADET;
S-R-01-0002-RQSERV-__ACK|000|
C-C-01-0002-DSCTRN-DLOAD|000|
S-R-01-0002-DSCTRN-__ACK|000|
....
```

Puis cliquer sur "sauvegarder le log" et donner un nom de fichier.

### Lecture du fichier trames CDM

Ce fichier peut être relu à n'importe quel moment, il permet de simuler l'enregistrement effectué.



## Arrêt des logiciels

Pour l'arrêt des logiciels, procéder dans l'ordre :

1. Arrêter le mode Run,
2. Arrêter le serveur Comm IP sur CDM Rail, ou fermer directement CDM rail
3. Fermer le programme client.

## Annexe - Liste des états d'un signal SNCF complet à 10 feux

signal hypothétique complet à 10 feux :



### Etats possibles de ce signal :

- |  |   |
|--|---|
| 1. Carré   | arrêt absolu  |
| 2. Sémaphore                                     | arrêt ou franchissable  |
| 3. Sémaphore clignotant                          | franchissable à 15km/h, peut annoncer un carré  |
| 4. Vert  | voie libre  |
| 5. Vert clignotant                               | limitation à 160km/h  |
| 6. Carré violet                                  | arrêt absolu sur voie de garage ou de service   |
| 7. Blanc   | départ en manoeuvre autorisé  |
| 8. Blanc clignotant                              | départ en manoeuvre limitée autorisé  |
| 9. Avertissement                                 | annonce un carré, un sémaphore, ou un sémaphore clignotant  |
| 10. Avertissement clignotant                     | annonce un avertissement ou un ralentissement implantés à faible distance                             |
| 11. Ralentissement 30                            | annonce un rappel 30  |
| 12. Ralentissement 60                            | annonce un rappel 60  |
| 13. Ralentissement 60 + avertissement clignotant | annonce un rappel 60 et un avertissement  |
| 14. Rappel 30                                    | franchissement d'aiguille à 30 km/h   |
| 15. Rappel 60                                    | franchissement d'aiguille à 60 km/h   |
| 16. Rappel 30 + avertissement                    | franchissement d'aiguille à 30 km/h et annonce un signal rouge ou un nouveau rappel de ralentissement |
| 17. Rappel 30 + avertissement clignotant         | franchissement d'aiguille à 30 km/h et annonce un avertissement                                       |
| 18. Rappel 60 + avertissement                    | franchissement d'aiguille à 60 km/h et annonce un signal rouge ou un nouveau rappel de ralentissement |
| 19. rappel 60 + avertissement clignotant         | franchissement d'aiguille à 60 km/h et annonce un avertissement                                       |