

TAM VALLEY DEPOT

MANUEL DE L'UTILISATEUR

DU QUADLN_S

V1.0

N3IX ENGINEERING

25 AOUT 2014

Traduction : Gilles COLLIN
1^{er} octobre 2014

Toute reproduction est sous droit du traducteur. Contacter gilles.cn@free.fr

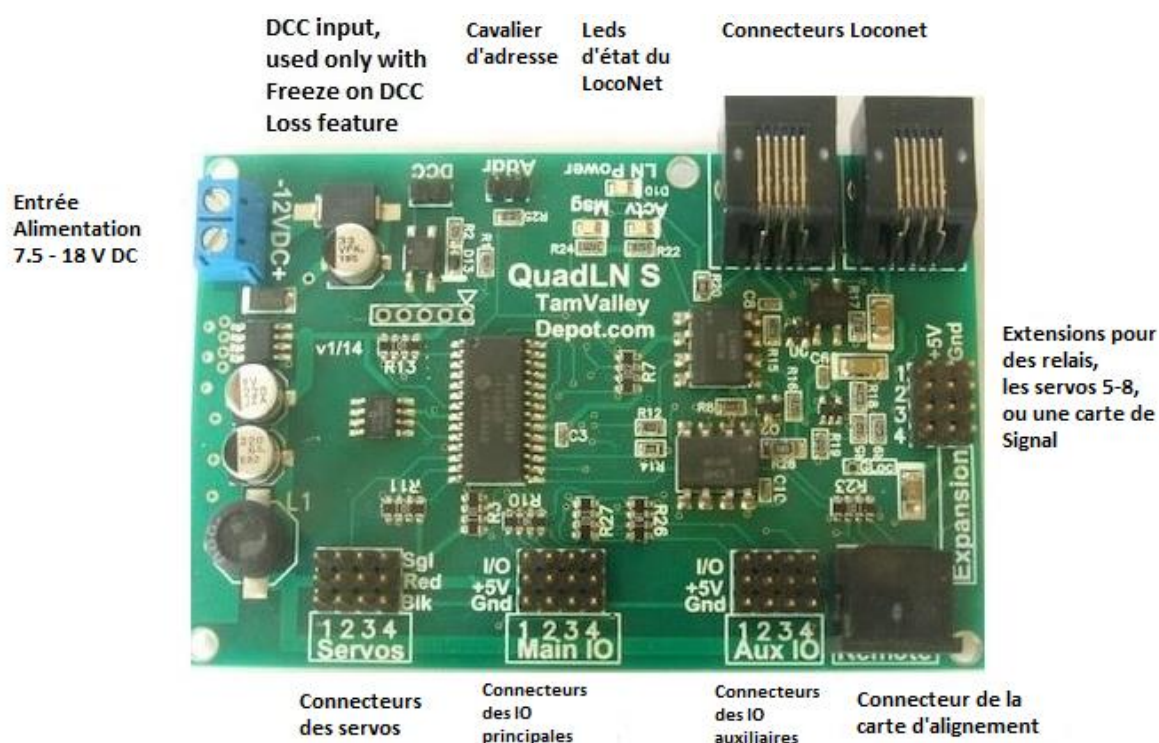
1 TABLE DES MATIERES

1	Table des matières	2
1	Bases.....	5
1.1	Vue d'ensemble.....	5
1.2	Servos	6
1.3	alimentation Externe.....	6
1.4	Indicateurs à led de la carte	6
1.4.1	Led d'alimentation du LN	6
1.4.2	Led de message du LN	6
1.4.3	Led d'activité du LN	7
1.5	Alimentation des coeurs.....	7
1.6	entrées.....	7
1.7	Adresses	7
1.7.1	Adresse de l'équipement.....	7
1.7.2	Adresse de début du servo.....	7
1.7.3	adresses de début verrouillée	8
1.7.4	Adresse de début des IO principales	8
1.7.5	Adresse de début des IO Aux	8
1.8	groupes.....	8
1.8.1	Indicateurs de TCO	9
2	Démarrage rapide.....	9
2.1	Connexion de l'alimentation	9
2.2	Configuration sans JMRI	9
2.2.1	Adresses de début des servos	9
2.2.2	Alignement manuel de la course et de la vitesse.....	9
2.2.3	Adresse de l'équipement.....	10
2.3	Configuration avec JMRI.....	10
2.3.1	Version minimum de JMRI	11
2.3.2	Création de la liste d'entrées et configuration de l'adresse d'équipement.....	11
2.3.3	Adresse de début de servo, adresse de debut de verrouillage, adresses de début des IO Main et Aux.....	12
2.3.4	Course et vitesse	12
2.4	Réinitialisation d'usine	12
2.4.1	Restauration des paramètres sélectionnés aux valeurs par défaut avec JMRI.....	13
2.4.2	Restauration de tous les paramètres aux valeurs d'usine en utilisant la carte de programmation	13
2.4.3	Restauration des paramètres sélectionnés à leur valeur d'usine sans utiliser JMRI	13
3	Contrôle de Servo.....	14
3.1	Course.....	14
3.1.1	Position fermée	14
3.1.2	Position déviée	14
3.1.3	vitesse.....	14
3.1.4	RapidStart.....	14
3.1.5	Vitesse directionnelle	15

3.1.6	Thrown speed	15
3.1.7	Thrown RapidStart.....	15
3.2	mode Verrouillage	15
3.2.1	Verrouillage local.....	16
3.2.2	Point milieu.....	17
3.3	Message – report de positions de servo sans capteurs	17
3.3.1	Exact Feedback (sim)	17
3.4	cascade	18
3.4.1	déclencheur	18
3.4.2	Action.....	18
3.4.3	turnout.....	18
3.4.4	Exemples de cascade	19
4	Lignes IO Main et Aux.....	20
4.1	Réponse d’une entrée	20
4.1.1	Type-Entrées Aux.....	20
4.1.2	Type-Entrées Main	20
4.1.3	Type Détecteur (points des IO Main seulement)	21
4.1.4	sensibilité (points des IO Main seulement).....	21
4.1.5	déclencheur	21
4.1.6	Blocage DCC.....	21
4.2	indication	22
4.2.1	Mode Led.....	22
4.2.2	Sens de la led	23
4.3	Action	23
4.4	Messages secondaires	23
4.4.1	type.....	24
4.4.2	Device	24
4.4.3	condition.....	24
4.4.4	number	24
4.4.5	Exemples de message secondaire	24
5	Routes.....	25
5.1	type.....	25
5.2	Entrées.....	26
5.2.1	Exemples de route.....	26
6	la fenêtre QuadLN_S	27
6.1	Adresses	27
6.1.1	Long Address	27
6.1.2	Servo Start Address	27
6.1.3	Lock Start Address	27
6.1.4	Main IO Start Address	27
6.1.5	Aux IO Start Address.....	28
6.2	Expansion Port.....	28
6.2.1	Route Delay	28
6.3	Servo OFF When Stopped.....	28
6.4	Servo State Memory.....	28

6.5	Signal State Memory	28
6.6	Retry on DCC Busy	29
6.7	Output Off Switch Request.....	30
6.8	No Command Station	30
6.9	Input Invert.....	30
6.10	Blink Indicates Unlock	31
6.11	Broadcast Local Actions.....	31
6.12	Device Information.....	31
6.12.1	Manufacturer ID	31
6.12.2	Product ID.....	31
6.12.3	Hardware Version.....	31
6.12.4	Software Version	32
6.12.5	Build Number.....	32
7	Mise à jour du logiciel	32
7.1	Effectuer la mise à jour.....	32
7.2	Mise à jour du modèle de décodeur	33
8	Tableau des CV	34
8.1	Tableau complet des CVs	34
8.2	Format de CV adresse longue.....	35
8.3	format des CV de configuration	35
8.4	format de CV de la configuration 2	35
8.5	format de CV de la vitesse de servo	35
8.6	Format de CV de position fermée du servo	35
8.7	Format de CV des adresses en cascade de servo	36
8.8	Format de CV de l'option IO	36
8.9	Format de CV de l'option détecteur sur IO Main – entrée Détecteur	36
8.10	Format de CV de l'adresse secondaire IO Main	36
8.11	Format de CV de l'Action d'IO	37
8.12	Format de CV des entrées de routes.....	37
8.13	Format de CV de contrôle de routes	37
8.14	Format de CV d'aspect de signal	37
8.15	Format de CV de configuration d'aspect de signal.....	38
8.16	Format de CV d'adressage d'aspect de signal – mode signal.....	38
8.17	Format de CV d'adressage d'aspect de signal – mode aiguillage.....	38
8.18	Format de CV de signal à lampe	38
8.19	Format de CV de signal à Led	38
9	notes du traducteur	39
9.1	connexion	39
9.2	déclaration.....	39
9.3	Programmation de l'adresse de la carte	40
9.4	Programmation des adresses de bases des accessoires	41
9.5	Programmation de chaque accessoire	42
9.6	tests.....	43

1 BASES



1.1 VUE D'ENSEMBLE

Le **QuadLN_S** est un décodeur stationnaire qui peut commander 4 servos pour contrôler des aiguillages. Les servos peuvent être actionnés en utilisant un bouton poussoir ou des entrées de commutation. Une seule sortie peut contrôler un ou plusieurs servos. Le **QuadLN_S** possède 3 configurations :

- 4 servos, 8 IO plus 4 sorties de relais externes (Mode Relai)
- 8 servos, 8 IO (Mode Octo)
- 4 servos, 8 IO plus 24 Leds en utilisant une carte de signal optionnelle

Le **QuadLN_S** peut afficher l'état du servo sur le panneau de contrôle de Tam Valley Depot ou un panneau à Led personnalisé de l'utilisateur. En plus de montrer si l'aiguillage est en position fermée ou déviée, les **QuadLN_S** peuvent indiquer quand un servo est en mouvement (sans nécessiter de capteurs de retour de position!) et quand il est verrouillé.

Le **QuadLN_S** fournit des effets de signalisation réalistes pour les signaux Searchlight et Color Position Light et pour les signaux de passage à niveau à l'aide de la carte de signal optionnelle. Les deux adressages de signaux Tête/mât et d'aiguillage pour les signaux sont disponibles.

Le **QuadLN_S** prend en charge jusqu'à 4 détecteurs de détection de courant ou des détecteurs optiques utilisant la carte de détecteur en option (dans le futur).

Le **QuadLN_S** est un équipement LocoNet. Les Servos peuvent être contrôlés par des commandes de commutation LocoNet. La Position de l'aiguillage peut être reportée à des logiciels tels que JMRI. Les

QuadLN_S fournit même un contrôle local de verrouillage de sorte que le bouton-poussoir local ou à bascule d'asservissement peuvent être désactivés sur une base d'asservissement individuel lorsque cela est désiré par simple envoi d'une commande Switch LocoNet.

Les Entrées du **QuadLN_S** peuvent être raccordées à des détecteurs externes et ensuite être reportées comme occupation de bloc en utilisant des messages généraux LocoNet de capteurs. Presque tous les appareils avec une fermeture de contact ou un signal de sortie de niveau logique 5V peuvent être raccordés à une entrée **QuadLN_S** disponible puis reportée par le LocoNet.

Le **QuadLN_S** peut également commander des sélections et des routes en cascade ce qui permet de tirer avantage du LocoNet pour implanter des fonctions de contrôle du réseau comme dans les dépôts ou des sélections de voies. Bien que la programmation de ces caractéristiques soit complexe plus que de configurer un simple aiguillage, l'utilisation des modèles dans JMRI simplifie ces configurations.

1.2 SERVOS

Le **QuadLN_S** contrôle jusqu'à 8 servos de moteur d'avion pour contrôle des aiguillages ou pour faire fonctionner d'autres tâches d'animation du réseau. Indépendamment du contrôle du mouvement et de la vitesse de chaque servo, avec une caractéristique RapidStart, elle permet un mouvement plus réaliste de l'aiguillage.

1.3 ALIMENTATION EXTERNE

Le **QuadLN_S** exige une alimentation externe DC entre +7.5V et +18V. Ne dépassez pas + 18 V DC. Le courant de consommation est de 100-200 mA à partir d'une alimentation 12V, mais les gros servos peuvent augmenter le courant au-delà de cette fourchette. Voir la section 2.1 pour les détails de branchement.

1.4 INDICATEURS A LED DE LA CARTE

Il y a 3 Leds sur le **QuadLN_S**.

1.4.1 LED D'ALIMENTATION DU LN

La Led d'alimentation du LN indique que le RailSync du LocoNet est présent et que l'interface optoélectronique d'isolation du LocoNet est alimentée. La Led est allumée quand un câble LocoNet est branché sur le **QuadLN_S**.

1.4.2 LED DE MESSAGE DU LN

La Led MSG est normalement allumée. La Led clignote quand un message LocoNet est vu, autre que les messages envoyés par le **QuadLN_S**. Si la Led ne clignote pas il doit y avoir un problème de câblage LocoNet.

1.4.3 LED D'ACTIVITE DU LN

La Led ACTV est normalement allumée. La Led ACTV clignote quand un message LocoNet est envoyé par le **QuadLN_S** ou quand le **QuadLN_S** réagit à un message LocoNet entrant. La Led clignote aussi quand le DCC change d'état.

1.5 ALIMENTATION DES COEURS

Le **QuadLN_S** peut commander des cartes de relais externes 5V pour alimenter les cœurs d'aiguillages ou pour d'autres utilisations. L'état du relais change au point milieu du servo pour réduire les problèmes de courts-circuits électriques pendant le mouvement avec certains aiguillages.

1.6 ENTREES

Le **QuadLN_S** possède 4 entrées principales et 4 entrées auxiliaires. Les 2 types d'entrées sont complètement configurables, comprenant quand elles répondent et quel sont les actions à prendre. Les actions possibles sur un changement d'entrée comprennent le contrôle des servos embarqués, l'envoi de commandes de commutation à d'autres décodeurs ou de déclencher des routes, et la génération de messages généraux de capteurs qui signalent le changement d'entrée sur n'importe quel équipement LocoNet concerné.

Le **QuadLN_S** possède des caractéristiques spéciales qui permettent aux lignes d'entrée de commander des Leds ou d'autres équipements. Depuis que les entrées ont des capacités de sorties, chaque connexion est appelée ligne IO dans ce manuel (IO = Input/Output).

1.7 ADRESSES

1.7.1 ADRESSE DE L'EQUIPEMENT

Le **QuadLN_S** nécessite une adresse d'équipement ainsi ces caractéristiques peuvent être configurées par la programmation en mode OPS du LocoNet. Cette adresse est juste là pour la programmation de l'équipement, et est indépendante des adresses des servos et des capteurs qui sont décrites ensuite dans les sections 1.7.2 et 1.7.3. La gamme des adresses d'équipement est de 0 à 16383 et doit être unique.

Typiquement une adresse étendue à 5 digits est utilisée pour l'adresse de l'équipement. Ceci évite des conflits avec les locomotives et les autres équipements quand on fait une programmation en mode OPS. L'adresse par défaut est 11000. Quand vous installez des **QuadLN_S**, rappelez-vous de ne brancher qu'un seul équipement non programmé à la fois au LocoNet et changez l'adresse par une adresse unique en suivant la procédure en 2.2.3 ou 2.3.2.

1.7.2 ADRESSE DE DEBUT DU SERVO

Le **QuadLN_S** répond aux commandes de commutateur LocoNet aux adresses commençant à l'adresse de début du servo. Le **QuadLN_S** utilise 4 ou 8 adresses consécutives de commutateur selon la configuration. L'adresse de début du Servo peut être de 1 à 2045. Évitez d'utiliser les adresses de la

plage 1017-1020 pour tout Servo si possible (ces adresses sont également utilisées par des commandes spéciales d'interrogation).

1.7.3 ADRESSES DE DEBUT VERROUILLEE

Le **QuadLN_S** verrouille les réponses aux commandes LocoNet de commutation à des adresses à partir de l'adresse de début verrouillage. Les servos **QuadLN_S** utilise 4 ou 8 adresses commutation consécutives en fonction de la configuration. L'adresse de verrouillage de démarrage peut être de 1 à 2045. Évitez d'utiliser les adresses de 1017 à 1020 pour tout Verrouillage si possible (ces adresses sont également utilisées pour des commandes spéciales d'interrogation). Si les plages d'adresses de servos et de verrouillage se chevauchent, tout verrouillage avec la même adresse de servo ne sera pas accessible en utilisant les commandes de commutation LocoNet.

1.7.4 ADRESSE DE DEBUT DES IO PRINCIPALES

Le **QuadLN_S** renvoie l'état de ses entrées principales en utilisant les adresses LocoNet à partir de l'adresse de début des IO principales. Le **QuadLN_S** utilise 4 adresses de capteurs consécutifs pour Main IO1 – Main IO4. L'adresse principale de début d'IO peut être de 1 à 4093. En général, la plage d'adresses d'IO principale ne doit pas chevaucher la plage d'adresse d'IO Aux.

1.7.5 ADRESSE DE DEBUT DES IO AUX

Le **QuadLN_S** renvoie l'état de ses entrées Auxiliaires en utilisant les adresses LocoNet à partir de l'adresse de début des IO Aux. Le **QuadLN_S** utilise 4 adresses de capteur successives, assignées à Aux IO1 - Aux IO4. L'adresse de début peut être de 1 à 4093. En général, la plage d'adresses d'IO principale ne doit pas chevaucher la plage d'adresse d'IO Aux.

1.8 GROUPES

Le **QuadLN_S** possède 4 groupes fonctionnels en mode Relais ou Signalisation. Chaque groupe comprend un servo, une ligne IO Main et une ligne IO Aux. A la livraison, les lignes IO sont configurées pour un bouton poussoir et change l'état du servo sur chaque pression sur le bouton.

Le **QuadLN_S** dispose de 8 groupes fonctionnels en mode Octo. Chaque groupe se compose d'un servo et soit d'une ligne IO principale ou d'une ligne IO Aux. Lors de la livraison, la ligne IO de chaque groupe est configurée pour un bouton poussoir momentané et change l'état du servo à chaque pression de touche.

Toutes les entrées peuvent être configurées pour contrôler tous les servos. En fait toutes les entrées peuvent contrôler les servos du **QuadLN_S** comme désiré, ou ils peuvent servir à d'autres fonctions sans contrôle des servos **QuadLN_S**. Il y a des caractéristiques du **QuadLN_S** (par exemple des indicateurs à Led) qui ne fonctionnent que lorsque l'entrée est utilisée avec le servo dans le même groupe, ce qui est la meilleure configuration quand vous voulez apprendre à vous servir du **QuadLN_S**.

1.8.1 INDICATEURS DE TCO

Le **QuadLN_S** peut alimenter des Leds sur un TCO connectées aux lignes IO MAIN et Aux. L'indication sur le TCO est le reflet des lignes spécifiques IO de l'état du servo dans le même groupe. Ainsi lors de l'utilisation des indicateurs sur un TCO telles que le Tam Valley Depot Fascia Controller, il est préférable d'utiliser une ligne IO dans le même groupe que le servo contrôlé ainsi les indications de position, de mouvement et de verrouillage sont significatives.

2 DEMARRAGE RAPIDE

2.1 CONNEXION DE L'ALIMENTATION

Le **QuadLN_S** exige une alimentation externe DC entre +7.5V et +18V. Ne dépassez pas les 18V. Le courant de consommation est de 100-200 mA à partir d'une alimentation 12V, mais les gros servos peuvent augmenter le courant au-delà de cette fourchette. Les entrées d'alimentation sont sur un bornier à vis du côté haut gauche de la carte. Connectez le fil – de l'alimentation sur le point le plus haut (Le plus proche du coin de la carte) et le fil + sur le point en bas (Le plus loin du coin de la carte). Si les fils sont inversés le QuadLN_S ne fonctionnera pas mais ne sera pas endommagé.

2.2 CONFIGURATION SANS JMRI

Tous les équipements sont contrôlés par des valeurs de CV ainsi il est possible de les configurer complètement manuellement. Généralement, ceci nécessite de déchiffrer les tableaux de CV de la section 8 et de calculer manuellement les bonnes valeurs, cependant les étapes suivantes vous permettent de programmer votre QuadLN_S et de le faire fonctionner pour contrôler des aiguillages.

2.2.1 ADRESSES DE DEBUT DES SERVOS

Programmez en premier l'adresse de début des servos pour éviter tout conflit avec d'autres décodeurs stationnaires. (Sauter cette étape si votre équipement n'est pas connecté au LocoNet)

- Positionnez le cavalier **ADDR**. Les Leds **ACTIVE** et **MSG** sur la carte clignotent alternativement.
- En utilisant votre manette, émettez une commande de commutateur avec l'adresse de début de servo. Quand la commande est reçue, les Leds s'arrêtent de clignoter et l'adresse est programmée.
- Enlevez le cavalier **ADDR**.

Les servos sont maintenant configurés avec 4 adresses consécutives commençant par l'adresse envoyée par la commande de commutation. Évitez d'utiliser des adresses 1017-1020 pour tout Servo si possible (ces adresses sont également utilisées pour des commandes spéciales d'interrogation).

2.2.2 ALIGNEMENT MANUEL DE LA COURSE ET DE LA VITESSE

Les extrémités de course et la vitesse pour chaque servo peuvent être réglées manuellement en utilisant les boutons **Select**, **UP** et **Down** à partir de la carte de réglage d'alignement. Voici la procédure de réglage.

Connectez la carte de réglage sur le **QuadLN_S**. Enlever le cavalier **ADDR** s'il est installé.

1. Appuyez sur le bouton Sélect jusqu'à ce que la Led sélection Servo s'allume (environ 1 seconde), indiquant que vous êtes en mode sélection de servo et que le servo indiqué est sélectionné. (le servo sélectionné est celui qui avait été sélectionné lors de la dernière action de réglage)
2. Utilisez les boutons UP et DOWN pour sélectionner le servo désiré, puis appuyez brièvement sur le bouton SELECT.
3. La Led CLOSED clignote maintenant pour indiquer que la position CLOSED est en cours de réglage. Utilisez les boutons UP et DOWN pour bouger le servo à la position désirée. Quand elle est réglée, appuyez brièvement sur le bouton SELECT.
4. La Led THROWN clignote maintenant pour indiquer que la position THROWN est en cours de réglage. Utilisez les boutons UP et DOWN pour bouger le servo à la position désirée. Quand elle est réglée, appuyez brièvement sur le bouton SELECT.
5. La Led SPEED clignote maintenant pour indiquer que la vitesse est en cours de réglage. Utilisez les boutons UP et DOWN pour changer la vitesse. Le servo changera de position pour démontrer la nouvelle vitesse. Vous pouvez continuer de régler la vitesse pendant le mouvement du servo. Quand elle est réglée, appuyez brièvement sur le bouton SELECT.
6. Pour réajuster un des paramètres, appuyez brièvement sur le bouton SELECT pour revenir à l'étape 4.

Pour sortir du mode réglage n'importe où, appuyez sur le bouton SELECT jusqu'à ce que les Leds s'éteignent (environ 1 seconde). Les nouveaux paramètres sont sauvés et peuvent être utilisés.

2.2.3 ADRESSE DE L'ÉQUIPEMENT

L'adresse de l'équipement peut être changée en utilisant les messages de programmation en mode SERVICE qui sont reçus par le QuadLN_S par le LocoNet. (Aucune connexion entre le QuadLN_S et la voie de programmation n'est nécessaire). Choisissez une adresse unique entre 1 et 16383. Voir la section 1.7.1 pour l'information sur l'adresse de l'équipement. Le format du CV est présenté dans le tableau 8-2 - Notez que les 2 bits supérieurs dans le CV17 doivent être 1. Une adresse unique 5 digit est recommandée pour éviter tout conflit.

- Enlever toutes les locomotives de la voie de programmation.
- Positionnez le cavalier **ADDR**. Les Leds **ACTV** et **MSG** doivent clignoter alternativement.
- Entrer en mode programmation et sélectionnez le mode Paginé.
- Entrez l'octet haut (les 8 bits supérieurs) de l'adresse au CV17.
- Entrez l'octet bas (les 8 bits inférieurs) de l'adresse au CV18.
- Enlevez le cavalier **ADDR**.

2.3 CONFIGURATION AVEC JMRI

Les modèles du **QuadLN_S** dans JMRI fournissent une manière plus facile et plus rapide pour configurer tout l'équipement. Cette section couvre les besoins basiques pour rendre le décodeur exploitable. Les sections plus en amont plonge dans les détails.

2.3.1 VERSION MINIMUM DE JMRI

Le modèle **QuadLN_S** utilise les fonctionnalités introduites dans la version 3.8 de JMRI. Si vous utilisez une version antérieure de JMRI, mettez à jour à la dernière version avant de poursuivre.

2.3.2 CREATION DE LA LISTE D'ENTREES ET CONFIGURATION DE L'ADRESSE D'EQUIPEMENT

Chaque décodeur nécessite sa propre liste d'entrée qui est stockée de manière unique pour chaque décodeur. Dans cette étape, nous voulons créer la liste des entrées pour le décodeur. Nous voulons également assigner une adresse d'équipement unique au décodeur en utilisant la programmation en mode service/voie de programmation ainsi nous pourrons plus tard modifier sa configuration à chaque fois que cela est nécessaire sans affecter les autres équipements. Quand on entrera tous les autres paramètres du **QuadLN_S**, nous utiliserons la programmation en mode OPS (opérationnel POM). La programmation en mode service/voie de programmation est seulement utilisée à cette étape et quand il y a eu une remise aux paramètres usine.

L'adresse peut être configurée en utilisant des messages de programmation en mode service/voie de programmation qui sont reçu par le QuadLN_S via le LocoNet. (Aucune connexion nécessaire entre le QuadLN_S et la voie de programmation).

- Enlevez toutes les locomotives de la voie de programmation.
- Dans DecoderPro ou PanelPro, cliquez sur le bouton de programmation **Service-Mode** ou Allez à **Programmers** dans l'onglet **Tools** et ouvrez le programmeur **Service-Mode**. Dans DecoderPro3 sélectionnez l'option **Programming Track** puis cliquez sur le bouton "+ New Loco".
- Parcourez la liste des décodeur jusqu'à **Tam Valley Depot**. Développez-la, sélectionnez le **QuadLN_S_v1** puis cliquez sur **Open Programmer**. (voir la note plus bas si l'entrée manque).
- Dans l'onglet **Roster Entry**, entrer un ID pour l'équipement comme QLNnnnnn (où nnnnn est l'adresse 5 digits de l'équipement) ou tout autre identificateur unique.
- Allez à l'onglet **QuadLN_S_S**.
- Sélectionnez **Use Long Address**.
- Entrez l'adresse souhaitée de l'équipement dans le champ **Long Address**. Choisissez une adresse unique entre 1 et 16383. Voir la section 1.7.1 pour plus d'informations. Une adresse unique 5 digits est recommandée pour éviter les conflits.
- Cliquez sur **Save** dans le menu **File** pour sauvegarder la nouvelle liste d'entrées.
- Allez à l'onglet **CVs**.
- Positionnez le cavalier **ADDR**. Les Leds **ACTV** et **MSG** doivent clignoter alternativement.
- Cliquez sur **WRITE CV17** puis sur **WRITE CV18**. (ignorez le message « programmer error », c'est normal).
- Enlevez le cavalier **ADDR**.

Dans DecoderPro ou PanelPro **Close** le **Programmer Service-mode**. Dans DP3 sélectionnez l'option **Program on Main**

Note : si les entrées **Tam Valley Depot** et la famille **QuadLN_S** ou le décodeur **QuadLN_S_S_v1** ne sont pas présentes dans la liste des décodeurs, suivez les étapes suivantes pour les télécharger et les installer sur JMRI.

Téléchargez le modèle le plus récent dans la zone fichier /Decoder files/Tam Valley Depot/folder dans le groupe Yahoo des utilisateurs de JMRI. Placez le fichier dans le répertoire décodeur sous User File Locations spécifié dans vos préférences d'emplacement des fichiers de JMRI. Placez le dossier TVD dans le sous-dossier XML/decoders dans User File Location spécifié dans votre JMRI

- Préférences File Locations. (Créez ce dossier s'il n'existe pas déjà.)
- Fermez le **Programmeur Service-mode** s'il est resté ouvert.
- Sélectionnez **Recreate Decoder Index** dans le menu **Debug**.
- Quand cette étape est terminée, retournez à la première étape de cette section.

2.3.3 ADRESSE DE DEBUT DE SERVO, ADRESSE DE DEBUT DE VERROUILLAGE, ADRESSES DE DEBUT DES IO MAIN ET AUX

Maintenant nous sommes prêts à entrer tous les autres paramètres du **QuadLN_S**. Pour faire cela dans JMRI, nous utilisons le programmeur Ops-Mode (mode opérations).

- Dans DecoderPro ou PanelPro, cliquez sur le bouton **Programmer OPS-Mode/Main-Track** ou Allez à **Programmers** dans l'onglet **Tools** et ouvrez le **Programmer OPS-Mode/Main-Track**. Dans DecoderPro3 sélectionnez l'option **Programming On Main**.
- Sélectionnez la liste des entrées **QuadLN_S** que vous avez créée depuis le roster et ouvrez-la.
- Allez dans l'onglet **QuadLN_S**.
- Entrez l'adresse de début de servo dans le champ **Desired servo Start Address**.
- (optionnel) entrez l'adresse de début du verrouillage dans le champ **Desired Lock Start Address**.
- (optionnel) entrez l'adresse de début de l'IO Main dans le champ **Desired Main IO Start Address**.
- (optionnel) entrez l'adresse de début de l'IO Aux dans le champ **Desired Aux IO Start Address**.
- Cliquez sur **Write changes en sheet** pour configurer le **QuadLN_S**.
- (optionnel) Cliquez sur **Read full sheet** pour relire les valeurs d'adresses et confirmez les paramètres.
- Cliquez sur **Save** dans le menu **File** pour enregistrer la liste des entrées.

2.3.4 COURSE ET VITESSE

Vous pouvez suivre la procédure de la section 2.2.2 au-dessus ou juste utiliser quand vous le voulez, le programmeur **Ops-Mode** dans JMRI, cliquez sur un des onglets Groupe et explorez l'utilisation des informations de la section 3.

2.4 REINITIALISATION D'USINE

Les paramètres du **QuadLN_S** peuvent toujours être réinitialisés à leurs valeurs par défaut d'usine. La réinitialisation d'usine est réalisée en émettant une valeur spéciale du CV8 (ID du fabricant) en utilisant des messages de programmation en **Service Mode** qui sont reçus par le **QuadLN_S** via le LocoNet. (Aucune connexion entre le **QuadLN_S** et la voie de programmation n'est nécessaire). Des choix de menu sont fournis dans le fichier du modèle QuadLN_S dans JMRI pour réaliser cette opération facilement en utilisant le Programmeur **Service-Mode/Programming-Track**. Une méthode alternative qui ne nécessite pas JMRI est décrite plus loin.

2.4.1 RESTAURATION DES PARAMETRES SELECTIONNES AUX VALEURS PAR DEFAULT AVEC JMRI

- Enlever toutes les locomotives de la voie de programmation.

Allez à **Programmers** dans l'onglet **Tools** et ouvrez **Service Mode Programmer**. (Dans DP3 sélectionnez juste l'option **Programming Track**.)

- Sélectionnez le roster **QuadLN_S** et ouvrez-le.
- Sélectionnez le menu **Reset** et cliquez sur **Factory Reset...**
- Sélectionnez **Restore All CVs...**, **Restore addresses...** ou **Restore Servo and Input CVs...** pour spécifier quels paramètres sont à restaurer.
- Installez le cavalier **ADDR**. Les Leds **ACTV** et **MSG** doivent clignoter alternativement.
- Cliquez sur **OK** pour effectuer la réinitialisation.
- Enlever le cavalier **ADDR**.
- Fermez par **Close** le **Service Mode Programmer**.
- Si vous choisissez « Restore Addresses », un cycle actionne le QuadLN_S pour que le changement prenne effet.

2.4.2 RESTAURATION DE TOUS LES PARAMETRES AUX VALEURS D'USINE EN UTILISANT LA CARTE DE PROGRAMMATION

- Enlevez l'alimentation du Quad-LN.
- Installez le cavalier **ADDR**.
- Connectez la carte de programmation sur le Quad-LN.
- Appuyez sur les boutons Up et Down en même temps.
- Réappliquez l'alimentation sur le Quad-LN.
- Continuez d'appuyer sur les boutons Up et down jusqu'à ce que les Led **ACTV** et **MSG** clignotent alternativement pendant 1 seconde puis qu'elles s'arrêtent de clignoter.
- Relâchez les boutons.
- Enlevez le cavalier **ADDR**.

2.4.3 RESTAURATION DES PARAMETRES SELECTIONNES A LEUR VALEUR D'USINE SANS UTILISER JMRI

- Enlevez toutes les locomotives de la voie de programmation.
- Installez le cavalier **ADDR**. Les Leds **ACTV** et **MSG** doivent clignoter alternativement.
- Entrez en mode programmation sur votre manette.
- Entrez une des valeurs suivantes sur le CV8 pour spécifier quels sont les paramètres à restaurer.
 - 8 pour restaurer tous les paramètres d'adresses, de servo et des entrées à leur valeur d'usine.
 - 9 pour restaurer toutes les paramètres d'adresses à leur valeur d'usine.
 - 10 pour restaurer tous les paramètres de servo et d'entrées à leur valeur d'usine.
- Enlevez le cavalier **ADDR**.

- Sortez du mode programmation sur votre manette.
- Si vous avez entré 9 sur le CV8 pour restaurer que les adresses, un cycle actionne le QuadLN_S pour que les changements prennent effet.

3 CONTROLE DE SERVO

Pour accéder aux paramètres suivants dans JMRI, utilisez toujours le programmeur [OPS-Mode/Programming On Main](#).

3.1 COURSE

Les paramètres de course permettent de régler les extrémités fermées et déviées, la vitesse, la caractéristique RapidStart pour chaque servo.

3.1.1 POSITION FERMEE

La position fermée peut être configurée de 0 à 2400, où 0 est entièrement en sens inverse des aiguilles d'une montre et 2400 est entièrement dans le sens des aiguilles d'une montre.

3.1.2 POSITION DEVIEE

La position déviée peut être configurée de 0 à 2400, où 0 est entièrement en sens inverse des aiguilles d'une montre et 2400 est entièrement dans le sens des aiguilles d'une montre.

3.1.3 VITESSE

La vitesse peut être paramétrée entre 0 et 63, où 0 est très lent et 63 très rapide. Une vitesse typique est 4. Quand vous réduisez la vitesse sur d'autres types de moteurs à mouvement lent en baissant la tension, vous réduisez également le couple appliqué. Avec les servos, la position est contrôlée continuellement pendant le mouvement ainsi le couple maximum est appliqué. Ceci donne des mouvements réguliers même avec une vitesse très lente programmée.

3.1.4 RAPIDSTART

Avec tous les moteurs à mouvement lent, il est généralement nécessaire de s'assurer que les aiguilles de l'aiguillage prennent bien leurs positions et qu'elles collent bien au rail. Quand le moteur est inversé, il y a un moment où aucun mouvement ne se produit au début, puis l'aiguille accélère ensuite d'un seul coup. Le temps nécessaire au démarrage est sensible à la vitesse des moteurs à mouvements lents et nuit au réalisme attendu avec ce type de moteurs. Dans certains cas, cet effet donne l'impression que l'action de commande de l'aiguillage demandée par l'opérateur ne marche pas, et alors il ré actionne les boutons ou la commande.

La caractéristique RapidStart est conçue pour minimiser le point mort et donner un mouvement plus réaliste aux aiguilles de l'aiguillage. Quand RapidStart est actif et que le servo est inversé, le QuadLN_S bouge le servo dans la première partie de la course à vitesse rapide puis à la vitesse programmée dans les paramètres de vitesse pour le reste du mouvement. Le mouvement initial à

vitesse rapide démarre le mouvement des aiguilles plus tôt, puis le mouvement final à vitesse lente crée l'effet de mouvement lent désiré.

Le paramètre RapidStart vous permet de programmer la partie de mouvement qui se fait à vitesse rapide. Les paramètres RapidStart disponibles sont **None** (aucun), **one-eighth travel**, (1/8 de la course), **One-quarter travel** (1/4 de la course), **Three-eighths travel** (3/8 de la course).

3.1.5 VITESSE DIRECTIONNELLE

Normalement, le **QuadLN_S** fait fonctionner un servo à la même vitesse quel que soit le sens de déplacement. Le **QuadLN_S** permet aussi pour chaque servo d'avoir des paramètres indépendants de vitesse (et de RapidStart) pour chaque direction de mouvement. Cette caractéristique peut être très utile quand on utilise un servo pour faire fonctionner un sémaphore ou pour des fonctions d'animation, où la vitesse dans une direction est différente de l'autre.

Le paramètre de vitesse directionnelle permet de programmer indépendamment des valeurs de vitesse et de RapidStart pour chaque direction de mouvement. Les paramètres disponibles de vitesse directionnelle sont **Disable** (inactif) et **Enable** (actif). Quand la vitesse directionnelle est active, le modèle JMRI affichera 2 paramètres supplémentaires : « Thrown speed » et « Thrown RapidStart ».

3.1.6 THROWN SPEED

Quand la vitesse directionnelle est active, le paramètre Thrown Speed programme la vitesse de mouvement dans la direction dévié. Les paramètres de vitesse dans la section 3.1.4 contrôlent toujours la vitesse de mouvement dans la direction fermé.

3.1.7 THROWN RAPIDSTART

Quand la vitesse directionnelle est active, le paramètre Thrown Speed RapidStart programme le comportement RapidStart dans la direction dévié. Les paramètres RapidStart dans la section 3.1.5 contrôle toujours la vitesse de mouvement dans la direction fermé.

3.2 MODE VERROUILLAGE

Le **QuadLN_S** dispose de deux modes de verrouillage optionnels, **Local Lockout** et **Midpoint**, qui sont décrits ci-dessous. Les Commutateurs LocoNet aux adresses **Lock Start Address** à **Lock Start Address** + 3 contrôle les verrouillages des Servo 1 à 4 Servo respectivement. Quand un commutateur de verrouillage est dévié, le mode de verrouillage sélectionné pour le servo correspondant est activé, et lorsque l'interrupteur de verrouillage est fermé, le mode de verrouillage pour le servo correspondant est désactivé.

Les paramètres disponibles de mode de verrouillage pour chaque servo sont **None**, **Local Lockout** et **Midpoint**. Lorsque la valeur est Aucun, l'interrupteur LocoNet à l'adresse de verrouillage n'a pas d'effet. Lorsqu'il est réglé à la section locale de verrouillage, les commandes locales pour l'aiguillage seront désactivés (bloqués) quand le commutateur de commande de verrouillage est réglé sur dévié. Lorsqu'il est réglé sur Milieu, le servo se déplacera à la position milieu quand le commutateur de

commande de verrouillage est réglé sur dévié et reviendra à la dernière position commandée lorsque la commande de verrouillage est réglé sur Fermé.

3.2.1 VERROUILLAGE LOCAL

Dans des systèmes comme CTC, certains aiguillages sont contrôlés à distance par l'aiguilleur. Ces aiguillages contrôlés à distance sont normalement verrouillés, ainsi ils ne peuvent pas être manœuvrés par quelqu'un du public par inadvertance. Quand quelqu'un souhaite manœuvrer un aiguillage localement, il doit d'abord obtenir l'autorisation de l'aiguilleur qui déverrouille l'aiguillage afin qu'il puisse le manœuvrer localement.

En plus du fait de supporter des opérations réalistes à partir de CTC, la possibilité de verrouiller un aiguillage est utile pour prévenir d'un changement par inadvertance. Par exemple, si on fait rouler un train pendant un salon, il est possible de verrouiller la ligne principale pour éviter des accidents face à face et causer un déraillement et une destruction des équipements.

La caractéristique de verrouillage local du **QuadLN_S** permet un contrôle local du déverrouillage de l'aiguillage et la capacité de fournir une indication locale de l'état du verrouillage de l'aiguillage. Verrouiller un servo de **QuadLN_S** équivaut à ce que les actions de contrôle local par les lignes Main et Aux (par exemple par des boutons, des commutateurs, etc.) ne changent pas directement la position du servo. Le servo continue de répondre normalement aux commandes de commutateur du LocoNet à son adresse, ceci comprend des commandes d'origine d'un second message où par des routes.

Exemple 1 : le mode de verrouillage du Servo 1 est défini à **Local Lockout** et l'adresse début de verrouillage du **QuadLN_S** est fixée à 13. Cela signifie que le commutateur de commande de verrouillage pour le Servo 1 est à l'adresse de commutateur LocoNet 13. Lorsque l'interrupteur 13 est dévié, le contrôle local du servo 1 est verrouillé, et lorsque le commutateur 13 est fermé le contrôle local des Servo 1 est déverrouillé.

Exemple 2 : le mode de verrouillage du Servo 1 est défini à **None** et l'adresse début de verrouillage du **QuadLN_S** est fixée à 13. Un changement de commutateur 13 à dévié ou fermé n'a pas d'effet sur le contrôle local du servo 1.

Le mécanisme de verrouillage du QuadLN_S supporte le déverrouillage à la fois par l'opérateur local ou par l'aiguilleur à volonté. L'opérateur local peut contrôler le verrouillage en émettant la commande de commutateur appropriée pour verrouiller le contrôle du commutateur avec une manette. Une approche plus intéressante pour l'opérateur local est de contrôler le verrouillage par un interrupteur ou une autre installation d'équipements électriques qui est connecté à un point d'entrée du Quad-LN. Il y a un exemple en 4.4.5 montrant comment le réaliser. Avoir un équipement local pour contrôler le verrouillage de cette manière est semblable à la réalité, mais nécessite un matériel comme un interrupteur à clé pour chaque aiguillage à déverrouiller.

Une autre approche est que l'aiguilleur contrôle le verrouillage en émettant la commande de commutateur appropriée, qui peut être faite à partir d'un levier du panneau de contrôle par exemple. Cette approche ne nécessite pas d'équipements supplémentaires, et le TCO Tam Valley peut indiquer l'état de verrouillage pour éviter toute confusion (voir la section 4.2.1).

Pour verrouiller des aiguillages spécifiques et protéger une circulation continue dans une exposition, un verrouillage de tout aiguillage peut être effectué à partir d'une manette. Les commutateurs peuvent faire partie d'une route, ce qui rend possible de verrouiller une série d'aiguillages par une simple action. Le verrouillage ou le déverrouillage peut être même déclenché par un bouton poussoir ou un interrupteur raccordé à une des entrées du Quad-LN, permettant un usage facile du verrouillage et du déverrouillage d'aiguillages. Il y a un exemple dans la section 4.4.5 montrant comment configurer un bouton poussoir pour verrouiller une route. Les contrôleurs Tam Valley Fascia peuvent afficher l'état du verrouillage ainsi il est pratique de voir si le réseau est sécurisé (voir la section 4.2.1).

3.2.2 POINT MILIEU

Quand **Midpoint** est sélectionné, positionner le commutateur de contrôle de verrouillage à dévié permet au servo de se mouvoir entre les positions fermée et déviée. Le servo reviendra à la dernière position commandée lorsque la commande de verrouillage est réglé sur Fermé. Ce mode peut être utile pour contrôler un sémaphore 3 positions, un aiguillage à 3 voies, et des effets spéciaux d'animation.

Les deux Leds sur le Contrôleur Tam Valley Depot Fascia seront allumées quand le servo correspondant est en position milieu. Si vous utilisez des Leds bicolores, les éléments Rouge et Vert sont commutés très rapidement de telle façon que l'œil ne voit pas de clignotement mais une lumière jaune. Le taux de clignotement est de 100 fois par seconde. Si vous utilisez un circuit personnalisé pour piloter un indicateur, le ton jaune peut être modifiée en changeant les niveaux actuels des Leds rouge et verte (souvent plus courant pour le Rouge / moins de courant pour le Vert est nécessaire).

3.3 MESSAGE – REPORT DE POSITIONS DE SERVO SANS CAPTEURS

Le QuadLN_S envoie des commandes de positions continuellement à chaque servo, donc il connaît toujours la position du servo. Ainsi le QuadLN_S peut fournir des messages de report de position de l'aiguillage basés sur la position du servo. Cette caractéristique est une manière efficace pour fournir une indication visuelle de l'état de l'aiguillage sur votre TCO sans aucuns capteurs !

Les choix de messages disponibles sont **No Output Feedback** et **Exact Feedback (sim)**. Si vous utilisez un panneau de contrôle JMRI, **Exact Feedback (sim)** vous permet d'intégrer les retours de position sur ce type de panneau. Si vous n'utilisez pas de logiciel qui affiche le report de positions d'aiguillage ou que vous n'en avez pas besoin pour votre réseau, alors sélectionnez **No Output Feedback** pour éliminer les messages LocoNet inutiles.

3.3.1 EXACT FEEDBACK (SIM)

Le report exact normalement utilise 2 commutateurs de fin de course qui sont montés sur l'aiguillage. Les messages de report sont envoyés quand les butées sont atteintes. Dans le mode **Exact Feedback (sim)**, le QuadLN_S envoie les mêmes messages de report mais basés sur la position du servo. Quand le servo commence à se déplacer, le commutateur de limite correspondant à la

position en cours reporte un état bas et quand le servo arrive à sa limite correspondant à sa position finale il reporte un état haut.

3.4 CASCADE

Une action en cascade correspond à une condition où la commande d'un aiguillage déclenche une commande sur un second aiguillage. Si le second aiguillage déclenche une commande vers un troisième aiguillage, etc., la commande est appelée en cascade. Une action en cascade fournit une manière très facile de mettre en œuvre une route dans un dépôt ou un croisement en envoyant une seule commande de commutateur LocoNet.

Les détails des paramètres de cascade sont expliqués plus bas, suivi d'exemples pour illustrer ces bases.

3.4.1 DECLENCHEUR

Le déclencheur est une commande de commutateur spécifique qui initie un événement en cascade. Quand la commande de déclenchement est reçue, la commande de commutateur en cascade est envoyée.

Les paramètres de déclenchement en cascade disponibles pour chaque servo sont **None**, **Closed**, **Thrown** et **Closed or Thrown**. Quand il est programmé à **Closed**, l'action en cascade est déclenchée chaque fois que le servo reçoit une commande de commutateur Closed. Quand il est programmé à **Thrown**, l'action en cascade est déclenchée à chaque fois que le servo reçoit une commande de commutateur Thrown. Quand il est programmé à **Closed or Thrown**, l'action en cascade est déclenchée à chaque fois que le servo reçoit toute commande de commutateur.

3.4.2 ACTION

L'action est un type de commande de commutateur qui est envoyée à l'aiguillage en cascade quand la condition de déclenchement est rencontrée.

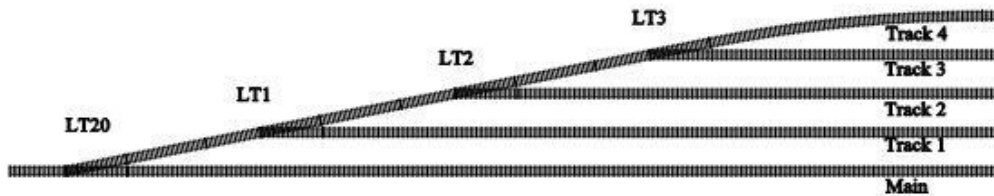
Les paramètres d'action en cascade disponibles pour chaque servo sont **Closed**, **Thrown**, **Follow** et **Invert**. Quand elle est programmée à **Closed**, une commande de commutateur Closed est envoyée. Quand elle est programmée à **Thrown**, une commande de commutateur Thrown est envoyée. Quand elle est programmée à **Follow**, une commande de commutateur correspondante à l'état du servo est envoyée. Quand elle est programmée à **Invert**, une commande de commutateur correspondante à l'état inverse du servo est envoyée.

3.4.3 TURNOUT

Le Turnout est le numéro de commutateur pour l'aiguillage qui doit recevoir la commande de commutateur en cascade.

3.4.4 EXEMPLES DE CASCADE

Alignement d'une échelle d'un triage. Nous avons des voies du triage accessibles par une simple échelle.



L'entrée du dépôt depuis la voie principale est contrôlée par l'aiguillage 20 : fermé pour la voie principale et dévié pour l'échelle. Les aiguillages du dépôt en échelle sont fermés pour le haut et déviés pour sélectionner une voie spécifique, ainsi l'aiguillage 1 est dévié pour la voie 1, l'aiguillage 2 est dévié pour la voie 2, etc. Supposez que nous voulions une seule route depuis la voie principale pour rejoindre n'importe quelle voie du triage avec une seule commande de commutateur.

En premier, programmez le déclencheur en cascade de l'aiguillage 1 à **Closed or Thrown**, l'action en cascade à **Thrown**, et le Turnout en cascade à 20. Maintenant à chaque fois que l'aiguillage 1 reçoit une commande, on émettra un message à l'aiguillage 20 de bouger en position déviée. Pour envoyer un train sur la voie 1, on ne devra plus que passer l'aiguillage 1 à dévié : l'aiguillage 20 passera à dévié par l'action en cascade.

Puis programmez le déclencheur en cascade de l'aiguillage 2 à **Closed or Thrown**, l'action en cascade à **Thrown**, et le Turnout en cascade à 1. Maintenant à chaque fois que l'aiguillage 23 reçoit une commande, on enverra un message à l'aiguillage 1 de bouger en position fermée. L'aiguillage 1 enverra une commande à l'aiguillage 20 de bouger en position déviée. Pour envoyer un train sur la voie 2, il suffit d'envoyer une commande déviée à l'aiguillage 2 : l'aiguillage 1 passera à fermé et l'aiguillage 20 passera à dévié par l'action en cascade.

Répétez ce processus pour chaque voie restante. La clef est que tout changement d'aiguillage qui sélectionne directement la voie envoie un message en cascade à l'aiguillage précédent qui les aligne comme il faut.

Croisement. Nous avons un croisement qui est contrôlé par 2 aiguillages aux adresses 6 et 15.



Pour des routes normales, les aiguillages 6 et 15 sont en position fermée. Pour le croisement, les aiguillages 6 et 15 sont en position déviée. Maintenant supposons que nous voulions simplement envoyer une commande à l'aiguillage 6 pour sélectionner les routes.

Une manière simple de réaliser ceci est que l'aiguillage 6 envoie un message en cascade à l'aiguillage 15 pour suivre dans tous les cas la position de l'aiguillage 6. Programmez le déclencheur en cascade de l'aiguillage 6 à **Closed or Thrown**, l'action en cascade à **Follow**, et le Turnout en cascade à 15. Maintenant à chaque fois que l'aiguillage 6 reçoit une commande, il enverra un message à l'aiguillage 15 de bouger à la même position.

4 LIGNES IO MAIN ET AUX

Chaque servo du QuadLN_S possède des lignes IO Main et Aux associées. Le QuadLN_S teste chaque ligne d'entrée 50 fois par secondes et filtre les changements de l'entrée pour supprimer les rebonds et le bruit électrique. L'utilisation la plus simple des possibilités d'une ligne IO d'entrée est de contrôler son servo associé. Des actions plus complexes sont également possibles, par exemple une seule ligne IO peut contrôler les 4 servos du Quad-LN. Le comportement spécifique de chaque entrée est configuré en utilisant les paramètres décrits dans cette section. Pour accéder aux paramètres suivants par JMRI, utilisez le programmeur **OPS-Mode/Programming On Main**.

4.1 REPOSE D'UNE ENTREE

4.1.1 TYPE-ENTREES AUX

Les types d'entrée disponibles pour chaque entrée auxiliaire sont **General** et **Detector**. Pour les entrées générales, un délai de stabilisation de 100 millisecondes est utilisé pour les deux fronts positifs et négatifs. Des options de déclenchement et de signalisation sont disponibles. Les appareils couramment connectés aux entrées générales sont des boutons, des fermetures de contact, et des sorties à collecteur ouvert.

Pour les entrées de détecteur, le déclencheur est toujours réglé sur les deux fronts et les options de déclenchement ne sont pas disponibles. Le temps de réponse pour les fronts montants est prolongé de 2 secondes pour s'assurer que l'indication occupé (bas) est maintenue lors de la perte momentanée du contact (détecteur de courant) ou d'un changement momentané du niveau de lumière (détecteur optique). Des options de signalisation sont proposées.

4.1.2 TYPE-ENTREES MAIN

Les types d'entrée disponibles pour chaque entrée principale sont **General** et **TVD Detector**. Pour les entrées générales, un délai de stabilisation de 100 millisecondes est utilisé pour les deux fronts positifs et négatifs. Des options de Déclenchement et de signalisation sont disponibles. Les appareils courants connectés aux entrées générales sont boutons, les fermetures de contact, et des sorties à collecteur ouvert.

L'entrée de détecteur TVD est utilisée lorsqu'il est connecté à une carte de détecteur Tam Valley Depot. Pour les entrées de détecteur TVD, le déclenchement est toujours réglé sur les deux fronts et

le temps d'anti-rebond pour les fronts positifs est étendu à 2 secondes pour veiller à ce que l'indication occupé (bas) soit maintenue lors de la perte momentanée du contact (détecteur de courant) ou de changement momentané du niveau de lumière (détecteur optique). Des options de type de détecteur et de sensibilité sont disponibles pour les entrées de détecteur TVD, alors que les options de déclencheur et de signalisation ne sont pas disponibles.

4.1.3 TYPE DETECTEUR (POINTS DES IO MAIN SEULEMENT)

Lorsque vous utilisez une carte de détection Tam Valley Depot, la détection par consommation de courant ou par infrarouge (IR) peut être attribuée individuellement à chaque broche d'entrée Main.

4.1.4 SENSIBILITE (POINTS DES IO MAIN SEULEMENT)

Quand un IO Main est réglé pour un Détecteur TVD, l'entrée est traité en utilisant une valeur de sensibilité afin de déterminer si le signal représente l'état occupé ou inoccupé. L'état occupé est signalé comme un 0 et l'état inoccupé est signalé comme un 1. Plus la valeur de la sensibilité est haute, plus sensible est l'entrée et plus la valeur du signal nécessaire pour la détection est bas. Réglez la valeur de la sensibilité juste assez haute pour assurer une détection fiable. Si une fausse détection se produit, diminuez la valeur de sensibilité.

4.1.5 DECLENCHEUR

Les actions des entrées sont initialisées seulement quand un changement sur l'entrée apparaît et qu'il correspond à la condition de déclenchement.

Les paramètres de déclenchement des entrées disponibles pour chaque entrée sont **None**, **Positive Edge**, **Negative Edge** et **Both Edges**. Quand elle est programmée à **Positive Edge**, l'action est déclenchée à chaque fois que l'entrée passe d'un niveau bas à un niveau haut. Quand elle est programmée à **Negative Edge**, l'action est déclenchée à chaque fois que l'entrée passe d'un niveau haut à un niveau bas. Quand elle est programmée à **Both Edges**, l'action est déclenchée à chaque fois que l'entrée change d'état. Quand elle est programmée à **None**, aucune action n'est déclenchée par l'entrée et la ligne de l'entrée n'est pas testée.

Pour les entrées **Detector**, l'entrée de déclenchement est fixée sur les deux fronts et la sélection du déclenchement n'est pas disponible.

Quand une ligne d'entrée surveillée change d'état et reste stable pendant la temporisation anti-rebond, un message Sensor LocoNet est généré.

4.1.6 BLOCAGE DCC

Certains équipements donnent de mauvaises indications quand le DCC n'est pas présent. Par exemple, un court-circuit dans une zone DCC peut engendrer sur les détecteurs d'occupation qu'il n'y a aucun train dans cette zone d'alimentation DCC ou qu'elle est vide. Le QuadLN_S a la possibilité de surveiller le signal DCC et de bloquer une entrée à son niveau antérieur quand le signal DCC est perdu, ce qui évite de faux reports des détecteurs pendant le court-circuit.

Les paramètres disponibles sont **Always Live** et **Freeze on DCC loss**. Quand il est programmé à **Always Live**, les actions des entrées suivent les paramètres de déclenchement indépendamment de l'état du DCC. C'est normalement le paramètre approprié quand l'entrée est utilisée pour le contrôle d'un servo. Quand il est programmé à **Freeze on DCC loss**, les actions des entrées sont inactives quand le signal DCC n'est plus présent. C'est normalement le paramètre approprié quand l'entrée est utilisée pour un détecteur d'occupation de canton.

4.2 INDICATION

Quand une ligne d'entrée n'est pas surveillée, le QuadLN_S peut traiter la ligne comme une sortie et peut alimenter une Led pour afficher l'état d'un servo. Comme l'échantillonnage est très rapide et plus qu'un œil puisse détecter, la Led apparait allumée sans voir le clignotement. (Si vous utilisez des équipements autres que les contrôleurs Tam Valley Depot Fascia, regardez le manuel pour câbler simultanément des interrupteurs et des Leds sur les lignes d'entrée).

Pour les Entrées de détecteur TVD la fonction d'indication à LED n'est pas disponible.

4.2.1 MODE LED

Quand l'indicateur à Led est actif, une Led fixe indique si le servo est fermé ou dévié. La Led peut être paramétrée pour clignoter rapidement pendant que le servo se déplace, ce qui donne un report d'indication sur la fin de course du servo quand la Led éclaire fixement. Quand le servo associé est verrouillable, la Led peut clignoter lentement pour donner une indication si le servo est verrouillé ou déverrouillé. (Le paramètre d'indication de verrouillage global détermine si le clignotement indique un état verrouillé ou déverrouillé – voir la section 6.8).

Pour les Entrées de détecteur TVD la fonction d'indication à LED n'est pas disponible.

Les paramètres du mode de la Led disponibles pour chaque entrée sont **Disable – Input only**, **Steady Drive**, **Blink on Move**, **Blink on Move or Lock**, **Blink on Lock**, **Lock State Position** et **Blink on Thrown**. Quand il est programmé à **Steady Drive**, l'indicateur de sortie montre la position du servo commandé. Quand il est programmé à **Blink on Move**, l'indicateur de sortie montre la position du servo commandé et clignote rapidement pendant le mouvement du servo. Quand il est programmé à **Blink on Move or Lock**, l'indicateur de sortie montre la position du servo commandé, clignote rapidement pendant le mouvement du servo et clignote lentement quand le servo est verrouillé et que l'état du verrouillage correspond aux paramètres de verrouillage. Quand il est programmé à **Blink on Lock**, l'indicateur de sortie montre la position du servo commandé et clignote lentement quand le servo est verrouillé et que l'état du verrouillage correspond aux paramètres de verrouillage. Quand il est programmé à **Lock State**, l'indicateur de sortie montre la position du servo commandé en verrouillage au lieu de sa position commandée. Quand il est programmé à **Position**, l'indicateur de sortie affiche la position actuelle du servo plutôt que sa position commandée, et il change seulement quand il atteint la position commandée et qu'il s'arrête. Quand il est programmé à **Blink on Thrown**, l'indicateur de sortie clignote quand le servo est commandée à la position déviée. Dans ce mode, les indicateurs des IO Main et Aux clignotent en opposition de phase, ce qui permet de les utiliser pour animer un passage à niveau quand un servo commande des barrières de passage à niveau.

Quand il est programmé à **Disable – Input only**, aucune sortie n'est disponible et la ligne reste en mode entrée. Ceci est nécessaire quand l'équipement d'entrée est un détecteur par exemple.

Sauf dans les modes **Steady** et **Position**, les sorties LED changent progressivement (fondus) d'un état à un autre pour simuler l'apparence d'une lampe à incandescence d'un panneau qui s'allume ou s'éteint. Dans les modes **Steady** et **Position** les sorties changent brusquement en une seule étape, de sorte que le signal de sortie peut être plus facilement utilisé pour déclencher d'autres types de dispositifs, si désiré.

4.2.2 SENS DE LA LED

Si le câblage de la Led sur le panneau donne le résultat inverse d'indication, le **QuadLN_S** peut tout simplement inverser la sortie de la Led. Ceci élimine la nécessité de recâbler la Led.

Pour les Entrées de détecteur TVD la fonction d'indication à LED n'est pas disponible.

Les paramètres de sens de la Led disponibles pour chaque entrée sont **Normal et Inverted**. Quand il est programmé à **Normal**, la sortie de Led est haute quand le servo est fermé et basse quand il est dévié. Quand il est programmé à **Inverted**, la sortie de Led est basse quand le servo est fermé et haute quand il est dévié.

4.3 ACTION

Chaque entrée du **QuadLN_S** peut contrôler chaque servo ou tous les servos. Ceci permet aux entrées de déclencher des routes locales sans avoir besoin de connexion LocoNet. L'action locale de chaque entrée des 4 servos du QuadLN_S peuvent être programmées indépendamment. (Une action locale sur un servo peut être effectuée si ce servo n'est pas verrouillé).

Les paramètres d'action de servo disponibles pour chaque entrée sont **None, Toggle, Follow, Invert, Closed, Thrown, Lock Toggle, Lock Follow, Lock Invert, Lock Closed and Lock Thrown**. Quand elle est programmée à **None**, l'état du servo correspondant n'est pas affecté par l'entrée. Quand elle est programmée à **Toggle**, l'état du servo correspondant est mis à l'opposé de son état actuel. Quand elle est programmée à **Follow**, l'état du servo correspondant est mis à fermé si l'entrée est haute et à dévié si l'entrée est basse. Quand elle est programmée à **Invert**, l'état du servo correspondant est mis à fermé si l'entrée est basse et à dévié si l'entrée est haute. Quand elle est programmée à **Closed**, l'état du servo correspondant est mis à fermé. Quand elle est programmée à **Thrown**, l'état du servo correspondant est mis à dévié. Quand un des paramètres de Verrouillage est sélectionné, l'état de verrouillage correspondant est mis en fonction.

4.4 MESSAGES SECONDAIRES

Chaque entrée peut recevoir et émettre un message secondaire. Ceci peut être employé pour implémenter plus de logique complexe pour des triages, des panneaux, et d'autres fonctions. Si elle est programmée pour envoyer un message secondaire, quand la condition de déclenchement de l'entrée configurée dans la section 4.1.1 est rencontrée, le message secondaire est envoyé. Si elle est programmée pour recevoir un message secondaire, un message de capteur LocoNet ou d'aiguillage qui correspond à un déclenchement secondaire d'actions de servo qui est configuré pour l'entrée.

4.4.1 TYPE

Les paramètres de type message secondaire disponibles pour chaque entrée sont **Trigger on this message** et **Send this message**. Quand il est programmé à **Trigger on this message**, les actions du servo spécifiée pour l'entrée sont exécutées quand la condition ci-dessous est rencontrée. Quand il est programmé à **Send this message**, le message secondaire est envoyé quand une condition de déclenchement d'entrée est rencontrée, et que le contenu du message est déterminé par la condition ci-dessous.

4.4.2 DEVICE

Les paramètres de Device de message secondaire disponibles pour chaque entrée sont **Turnout** et **Sensor**. Quand il est programmé à **Turnout**, le message secondaire est un message de commande d'aiguillage. Quand il est programmé à **Sensor**, le message secondaire est un message de capteur général.

4.4.3 CONDITION

Les paramètres de condition de message secondaire pour chaque entrée sont **None**, **Hi/Closed**, **Lo/Thrown** and **Both/Follow**.

Si le type est **Trigger on this message** : Quand il est programmé à **Hi/Closed**, la condition de déclenchement est un capteur à l'état haut ou un message fermé de commutateur. Quand il est programmé à **Lo/Closed**, la condition de déclenchement est un capteur à l'état bas ou un message fermé de commutateur. Quand il est programmé à **Both/Follow**, la condition de déclenchement est un capteur ou un message de commutateur avec un état correspondant à l'état de l'entrée. Quand il est programmé à **None**, la fonction de message secondaire est désactivée.

Si le type est **Send this message** : Quand il est programmé à **Hi/Closed**, le message envoyé est un capteur à l'état haut ou un message fermé de commutateur. Quand il est programmé à **Lo/Closed**, le message envoyé est un capteur à l'état bas ou un message fermé de commutateur. Quand il est programmé à **Both/Follow**, le message envoyé est un capteur ou un message de commutateur avec un état correspondant à l'état de l'entrée. Quand il est programmé à **None**, la fonction de message secondaire est désactivée.

4.4.4 NUMBER

Le Commutateur ou le Détecteur numérotent le message Secondaire.

4.4.5 EXEMPLES DE MESSAGE SECONDAIRE

Tous les aiguillages de la voie principale à fermé. Supposez que nous ayons plusieurs aiguillages de la voie principale qui sont contrôlés par des cartes Quad-LN. Chaque aiguillage est contrôlé par un bouton poussoir sur la ligne IO Main. Nous désirons mettre tous les aiguillages dans la position fermée avec une seule commande de commutateur, disant Switch 200 Closed. Pour chaque ligne IO main de servo, programmez le type de message secondaire à **Trigger on this message**, le Device à

Turnout, la condition à **Hi/Closed**, et le Number à **200**. Maintenant l'envoi d'une seule commande de commutateur aligne tous les aiguillages à la position fermée.

Tous les aiguillages de la voie principale à fermé avec un seul bouton poussoir. C'est la même chose que dans l'exemple précédent, sauf qu'on veut appuyer sur un bouton poussoir pour aligner les aiguillages plutôt que d'envoyer une commande de commutateur avec une manette. Faites la même programmation que dans l'exemple précédent. Câblez le bouton poussoir à une entrée IO Aux libre sur une carte Quad-LN, puis ajustez les paramètres pour la ligne IO Aux : déclenchement à **Positive Edge** ; action à **None** pour chaque servo ; type de message secondaire **Send this message**, Device à **Turnout**, condition à **Hi/Closed** et le nombre à **200**. Maintenant une pression sur le bouton poussoir envoie une commande fermé au commutateur 200, qui provoque de mettre en position fermée tous les aiguillages de la voie principale.

Déverrouillage d'un aiguillage de voie principale avec une touche. Nous avons un bouton poussoir local configuré pour contrôler l'aiguillage 17, mais nous désirons verrouiller le bouton poussoir jusqu'à ce qu'un interrupteur soit mis en œuvre. Câblez l'interrupteur sur une ligne IO libre sur une carte Quad-LN, puis ajustez les paramètres pour ligne IO comme suit : déclencheur à **Both Edges** ; action à **None** pour chaque servo ; type de message secondaire à **Send this message**, device à **Turnout**, condition à **Both/Follow** et le nombre à 21 (21 est le commutateur de contrôle de verrouillage pour l'aiguillage 17 – voir la section 3.2). Maintenant, l'ouverture de l'interrupteur envoie une commande de fermeture au commutateur 21 qui déverrouille l'aiguillage 17, et la fermeture de l'interrupteur envoie une commande déviée au commutateur 21 qui verrouille l'aiguillage 17. Un indicateur approprié sur le TCO peut surveiller le commutateur 21net fournir une indication quand l'aiguillage 17 est déverrouillé.

5 ROUTES

Les routes sont des séquences arbitraires de commutateurs et de message de capteur qui sont envoyés quand la condition de déclenchement est rencontrée. Le QuadLN_S possède 4 routes, chacune pouvant contenir 8 entrées de commande de commutateur ou de messages d'état de capteurs. Les routes peuvent être étendues à partir d'une route préalable, ainsi il est possible d'avoir une seule route à 32 entrées. Cette caractéristique peut permettre de faire des choses simples comme l'alignement d'aiguillages sur une voie principale. Pour accéder aux paramètres suivants par JMRI, utilisez le programmeur [OPS-Mode/Programming On Main](#).

5.1 TYPE

Les types de routes disponibles sont **None**, **Normal Route**, **Selector** and **Expand (Prior) Route**. Une **Normal Route** est seulement déclenchée par la première entrée et elle envoie toujours les entrées restantes comme elles sont entrées. Une **Selector Route** est déclenchée par n'importe quelle entrée et elle envoie les entrées restantes dans l'état opposé à celles entrées. Cette action implémente un 1 à N sélections. Une **Expanded Route** ajoute simplement ses entrées dans la route préalable.

5.2 ENTREES

Chaque entrée de route consiste à une adresse d'équipement et un type d'équipement/action. Pour les messages de commutateur les actions disponibles sont fermées ou dévié, pour les messages de capteur les actions disponibles sont capteur à l'état haut et capteur à l'état bas.

5.2.1 EXEMPLES DE ROUTE

Aligner une route quand un bouton est appuyé. Nous voulons être capables d'aligner une route en appuyant sur un bouton poussoir. Câblez le bouton poussoir à une ligne IO Aux libre sur une carte Quad-LN, puis ajustez les paramètres pour la ligne IO Aux pour un message programmé à **General Sensor**. Mettez l'action à **None** pour chaque servo. Choisissez une route dans un QuadLN_S et mettez le type à **Normal Route**, mettez la première entrée à **Aux IO Address** et à **Sensor Hi**, puis entrez **Switch address** et **Closed** ou **Thrown** dans les entrées suivantes pour chaque aiguillage de la route. Utilisez les routes étendues si nécessaire. Maintenant en appuyant sur le bouton on aligne la route entière.

Verrouillage de tous les aiguillages de la voie principale. Nous avons beaucoup d'aiguillages sur la voie principale contrôlés par des cartes Quad-LN. Chaque aiguillage est contrôlé par un bouton poussoir sur sa propre ligne IO Main. Nous voulons pouvoir verrouiller tous les boutons poussoir (pendant une expo par exemple) en envoyant une seule commande de commutateur, qui sera commutateur 200 fermé. Choisissez une route dans un QuadLN_S et mettez le type à **Normal Route**, mettez la première entrée à **200 Closed**, et ensuite entrez **Lock address** et **Thrown** dans les entrées suivantes pour chaque aiguillage à verrouiller de la voir principale. Utilisez les routes étendues si nécessaire. Maintenant en envoyant une seule commande de commutateur on verrouille tous les aiguillages de la voie principale.

Indicateur de sélection de voie sur un TCO. Nous avons un TCO pour un triage avec une Led pour chaque voie. Les Leds sont contrôlées par un décodeur stationnaire qui utilise des commandes de commutateur. L'envoi d'une adresse de commutateur particulière à fermé met la Led à On, et l'envoi de cette adresse à dévié met la Led à Off. Choisissez une route dans un QuadLN_S et mettez le type à **Selector** et ensuite entrez la **Switch address** et **Closed** dans les entrées de la route pour chaque adresse de commutateur qui contrôle une Led. Utilisez les routes étendues si nécessaire. Maintenant en envoyant une seule commande de commutateur, la Led s'allume pour la voie sélectionnée et les Leds s'éteignent pour toutes les voies qui ne sont pas sélectionnées.

Paramétrer des Signaux pour un aiguillage. Nous avons un certain nombre de signaux de protection du commutateur 200 et souhaitons mettre leurs aspects seulement sur la base de la position d'un aiguillage. Choisissez un itinéraire dans les **QuadLN_S** qui contrôle les signaux et sélectionnez le type **Normal Route**, placez la première entrée à **200 Closed**, puis entrez l'**Aspect Number** souhaité pour chaque signal et l'**Aspect** dans les entrées suivantes dans la route. Utilisez les Routes étendues selon les besoins. Puis répétez ces étapes pour l'autre **Normal Route**, cette fois avec **200 Thrown** comme première entrée et l'**Aspect Number** souhaité dans chacune des entrées suivantes. Maintenant, l'envoi d'une commande au commutateur 200 permettra sur chacun des signaux d'afficher l'aspect approprié.

6 LA FENETRE QUADLN_S

Le **QuadLN_S** possède un nombre de paramètres d'adresses et de paramètres de configuration spéciaux. La plupart des utilisateurs ont seulement besoin de programmer l'adresse de début des servos et l'adresse de début des capteurs (l'adresse longue est donnée seulement pour information et elle sera entrée si nécessaire). Les configurations spéciales restantes peuvent généralement être laissées à leurs paramètres par défaut. Pour accéder aux paramètres suivants par JMRI, utilisez le programmeur **OPS-Mode/Programming On Main**.

6.1 ADRESSES

6.1.1 LONG ADDRESS

L'adresse longue est l'adresse utilisée quand vous programmez l'équipement en **OPS-Mode**. Cette adresse est aussi affichée sur la fenêtre de base. Voir la section 1.7.1 pour plus d'informations. Une procédure spéciale est nécessaire pour changer l'adresse de l'équipement, et cette procédure est décrite dans la section 2.2.5. Cette procédure est conçue pour prévenir de changements d'adresse d'équipement par inadvertance.

6.1.2 SERVO START ADDRESS

Les adresses de commutateur pour les servos du QuadLN_S sont programmées en utilisant ce champ. Le **QuadLN_S** utilise 4 adresses consécutives pour commander les servos. Évitez d'utiliser les adresses de la plage 1017-1020 pour tout Servo si possible (ces adresses sont également utilisées par des commandes d'interrogation spéciales). L'adresse actuelle pour chaque servo est affichée dans l'onglet Group. Voir la section 1.7.2 pour plus d'informations.

6.1.3 LOCK START ADDRESS

Les adresses des commutateurs des verrouillages du QuadLN_S sont définies en utilisant ce champ. Le QuadLN_S utilise 4 adresses de commutations consécutives pour le contrôle du verrouillage. Évitez d'utiliser des adresses 1017-1020 pour tout Verrouillage si possible (ces adresses sont également utilisées pour des commandes d'interrogation spéciales). L'adresse de chaque Verrou est affichée dans l'onglet Group. Voir la section 1.7.3 pour plus d'informations sur l'adresse de début de verrouillage. Si les plages d'adresses de servo et de verrouillage se chevauchent, les verrous avec la même adresse qu'un servo ne seront pas accessibles via une commande Switch LocoNet.

6.1.4 MAIN IO START ADDRESS

Les adresses IO Main pour les servos QuadLN_S sont définies en utilisant ce champ. Le QuadLN_S utilise 4 adresses de capteurs consécutives pour les IO Main. Voir la section 1.7.4 pour plus d'informations sur l'adresse de début d'IO Main.

6.1.5 AUX IO START ADDRESS

Les adresses IO Auxiliaires pour les servos QuadLN_S sont définies en utilisant ce champ. Le QuadLN_S utilise 4 adresses de capteurs consécutives pour les IO Aux. Voir la section 1.7.4 pour plus d'informations sur l'adresse de début d'IO Aux.

6.2 EXPANSION PORT

Le port d'extension peut être utilisé pour piloter directement les cartes relais 5 Volt de Tam Depot Valley, 4 servos supplémentaires, ou 24 Leds à l'aide de la carte de signaux Tam Valley Depot. Ce paramètre détermine quels signaux sont fournis aux ports d'extension. Lorsqu'il est réglé à **Relays 1-4**, les signaux de commande de la carte relais sont fournis au changement d'état lorsque le servo est à mi-parcours. Lorsqu'il est réglé à **Signaling**, les signaux de commande de Leds sont fournis pour contrôler 24 Leds en utilisant la carte de signaux de Depot Tam Valley. Lorsqu'il est réglé à **Servos 5-8**, les signaux de commande d'asservissement sont fournis pour 4 servos supplémentaires.

6.2.1 ROUTE DELAY

Chaque action dans une route locale peut se produire immédiatement après que la route soit déclenchée ou peut se produire après un délai fixe. La durée du retard est commandée par ce paramètre. Les réglages **Route Delay** disponibles sont Retard Route **1 Sec**, **2 Sec**, **4 Sec** et **8 Sec**. La valeur par défaut est **1 Sec**.

6.3 SERVO OFF WHEN STOPPED

Ce paramètre envoie un signal de commande à un servo pour l'éteindre lorsque le servo ne se déplace plus. Dans les cas où peu de force est nécessaire pour maintenir l'aiguillage à la position désirée, comme dans le cas d'un ressort de maintien de l'aiguillage en position, l'extinction de la commande permettra d'éviter au servo de bourdonner. La valeur par défaut est **Unchecked** signifiant qu'il est toujours à On. Cochez la case pour activer l'option **Servo OFFWhen Stopped**.

6.4 SERVO STATE MEMORY

Ce paramètre détermine si le QuadLN_S enregistre les positions des Servo quand ils changent, puis restaure tous les servos à leur dernière position connue au démarrage. Par défaut, la case est **Checked** ce qui signifie que l'état de chaque servo sera restauré à la mise sous tension. Si la case n'est pas cochée, alors chaque Servo sera mis à Fermé au démarrage.

6.5 SIGNAL STATE MEMORY

Ce paramètre détermine si le QuadLN_S enregistre l'aspect des signaux quand ils changent et restaure tous les signaux de leur dernier aspect connu à la mise sous tension. Par défaut, la case est **décochée** ce qui signifie que toutes les sorties de signaux seront Off au démarrage. Si la case est cochée, l'état de chaque sortie de signal sera à la mise sous tension.

6.6 RETRY ON DCC BUSY

Ce paramètre contrôle si le QuadLN_S tente de renvoyer les requêtes de commutation lorsque la station de commande reporte que le bus DCC est occupé. La valeur par défaut est coché ce qui veut dire que les requêtes de commutation ont réémises en cas de besoin.

Remarque: Cette section est un peu technique. Les utilisateurs qui n'ont pas besoin d'une requête de commutation envoyée par les QuadLN_S pour atteindre les décodeurs stationnaires seulement DCC peut simplement ignorer cette section. Les informations ci-dessous sont incluses afin que les utilisateurs avec des configurations complexes qui incluent les décodeurs stationnaires seulement DCC peuvent mieux comprendre les compromis de performance du système.

Lorsque le QuadLN_S envoie une requête de commutation sur le LocoNet dans le cadre d'une cascade, d'un message secondaire ou d'une Route, la station de commande tente d'envoyer la commande sur le bus DCC. Ceci est fait de sorte que seuls les décodeurs stationnaires DCC recevront la commande. Cependant, en raison de la différence de bande passante, la requête de commutation LocoNet peut arriver à la station de commande plus rapidement que la station de commande ne peut les envoyer sur le DCC. Lorsque cela arrive, la station de commande met en mémoire tampon interne les requêtes de commutation afin de se donner une chance de les envoyer sur le bus DCC. Lorsque la station de commande reçoit une requête de commutation et que sa mémoire tampon interne est déjà pleine (ou que l'alimentation de voie sur les stations de commande est désactivée et que les commandes DCC ne peuvent pas être envoyées), la station de commande répond par un message spécial LocoNet pour indiquer que la requête de commutation n'a pas été transmise au bus DCC.

Quand une requête de commutation n'atteint pas le bus DCC, le QuadLN_S possède deux options: ignorer l'échec ou envoyer à nouveau la requête de commutation. L'envoi de la requête de commutation à nouveau normalement fonctionne bien, mais s'il y a un grand nombre de requêtes de commutation générés sur le LocoNet (à partir d'un ou plusieurs équipements), alors les tentatives peuvent commencer à s'accumuler. Lorsque cela se produit les performances se dégradent jusqu'à la station de commande ne soit plus en mesure de les acheminer, et dans la plupart des cas les choses se régler sans impact notable.

Si vous avez besoin que le QuadLN_S puisse envoyer des requêtes de commutation à des décodeurs stationnaires seulement DCC alors cochez cette option et les QuadLN_S réémettront les requêtes de commutation en cas de besoin. Si vous n'avez pas utilisé les QuadLN_S pour envoyer des requêtes de commutation à des décodeurs stationnaires seulement DCC alors décochez cette option, car dans ce cas ce n'est pas grave si une commande envoyée par le QuadLN_S n'arrive pas au bus DCC et donc il n'est donc pas nécessaire de créer du trafic LocoNet supplémentaire en renvoyant les requêtes de commutation.

Enfin, si vous utilisez un DCS100 et n'avez pas de décodeurs stationnaires seulement DCC du tout, vous pouvez fixer l'OPSW 27 à Fermé. La Fermeture de l'OPSW 27 demande au DCS100 de ne pas envoyer de requêtes de commutation sur le bus DCC. Ceci élimine toutes les questions de gestion de congestion et permet aux requêtes de commutation d'être envoyées à la pleine vitesse du LocoNet dans toutes les conditions.

6.7 OUTPUT OFF SWITCH REQUEST

Ce paramètre détermine si le QuadLN_S envoie une requête de commutation de sortie Off après une requête de commutation de sortie On. La valeur par défaut est Coché ce qui signifie d'envoyer requête de commutation de sortie Off.

Remarque: Comme pour la section précédente, cette section est un peu technique et la plupart des utilisateurs peuvent l'ignorer.

Lorsque le QuadLN_S envoie une requête de commutation sur LocoNet dans le cadre d'une cascade, d'un message secondaire ou d'une route, deux messages LocoNet sont générés. Le premier message est une requête de commutation et la seconde est une requête de commutation de sortie Off. Par convention, ces deux messages sont envoyés parce que certains décodeurs stationnaires mettent leurs pilotes de sortie à On quand ils reçoivent une requête de commutation de sortie et à Off quand ils reçoivent une requête de commutation de sortie Off. Le QuadLN_S insère un retard d'environ 100 millisecondes entre les deux messages.

Lors de la réception des requêtes de commutation, le QuadLN_S ignore les requêtes de commutation de sortie Off et laisse à On la sortie tout le temps. Beaucoup de décodeurs de moteur à mouvement lent agissent de cette façon. Bien que l'envoi de requête de commutation de sortie Off n'affecte pas les décodeurs il double le trafic LocoNet et double le trafic DCC (voir section 6.5) pour chaque requête de commutation. Cela peut réduire les performances brièvement pendant une rafale de requêtes de commutation, comme lors de l'envoi d'une longue route. Si tous vos décodeurs stationnaires ignorent les requêtes de commutation de sortie Off alors désactivez cette option pour éliminer les messages LocoNet inutiles.

6.8 NO COMMAND STATION

Ce paramètre est applicable aux installations seulement avec un LocoNet où aucune station de commande n'est présente. Les QuadLN_S peuvent étendre les réponses à certains messages de programmation pour fournir également les réponses normalement envoyées par une station de commande. Quand aucune station de commande n'est présente, ces réponses supprimeront les messages d'erreur dans JMRI ou d'autres logiciels qui s'attendent à voir les réponses de la station de commande. La valeur par défaut est décochée signifiant qu'une station de commande est présente. Cochez cette option si aucune station de commande n'est présente sur le LocoNet.

6.9 INPUT INVERT

Ce paramètre permet d'inverser la polarité du capteur d'entrée pour être utilisé avec certains programmes de contrôle de réseau qui nécessitent des entrées basses actives pour représenter un niveau haut. La valeur par défaut est décochée signifiant que les entrées ne sont pas inversées. Lorsqu'elle est cochée, le niveau du signal est inversé. Notez que les paramètres des entrées peuvent avoir besoin d'être ajustés à chaque fois cette option est modifiée.

6.10 BLINK INDICATES UNLOCK

Ce paramètre détermine si les indicateurs à Leds des lignes IO configurés pour afficher le statut de verrouillage du Servo clignotent lorsque le servo est à l'état bloqué ou quand il est dans l'état débloqué. La valeur par défaut est décochée et un clignotement de la LED indique l'état fermé. Lorsqu'elle est cochée un clignotement de la LED indique l'état déverrouillé.

6.11 BROADCAST LOCAL ACTIONS

Ce paramètre détermine si le QuadLN_S rend visible sur le LocoNet les actions locales de servo et de verrouillage. La valeur par défaut est cochée ce qui signifie que les Actions locales sont diffusées.

Les QuadLN_S peuvent effectuer des actions locales de servo et de verrouillage qui sont déclenchées par des changements d'entrée (voir la section 4.3 pour obtenir des informations sur les actions locales) de deux façons légèrement différentes. Dans la première approche, le QuadLN_S change directement les servos et les verrouillages à de nouvelles valeurs. Dans la deuxième approche, le QuadLN_S génère des commandes de requêtes de commutation sur le LocoNet comme si un autre équipement LocoNet envoyait les commandes. La seconde approche est utilisée par défaut et fournit au LocoNet la visibilité des changements d'état causés par les actions locales de la même façon que n'importe quelle requête de commutateur LocoNet. La seconde approche permet également aux changements d'entrée du QuadLN_S de déclencher des routes. Si la visibilité sur le LocoNet n'est pas nécessaire et que les routes locales ne sont pas utilisées, vous pouvez décocher l'option et éliminer le trafic de messages LocoNet inutile. Quel que soit le paramètre que vous choisissiez, les actions locales seront effectuées si le LocoNet est connecté.

6.12 DEVICE INFORMATION

Cette section contient les informations très pratiques pour résoudre des problèmes et faire des mises à jour.

6.12.1 MANUFACTURER ID

L'ID de fabricant de Tam Valley Depot est 59. L'écriture de valeurs spéciales sur ce paramètre quand il est en mode service peut réinitialiser l'équipement – voir la section 2.4.

6.12.2 PRODUCT ID

Cette valeur identifie le produit et vous aide à ne pas installer de mises à jour de logiciels incorrectes. L'ID du produit QuadLN_S est 5.

6.12.3 HARDWARE VERSION

Cette valeur identifie la révision matérielle du produit et vous aide à ne pas installer de mises à jour de logiciels incorrectes.

6.12.4 SOFTWARE VERSION

Cette valeur identifie la révision logicielle du produit et vous aide à ne pas installer de mises à jour de logiciels incorrectes.

6.12.5 BUILD NUMBER

Cette valeur est un identificateur logiciel additionnel et permet un meilleur support du constructeur.

7 MISE A JOUR DU LOGICIEL

7.1 EFFECTUER LA MISE A JOUR

La conception du QuadLN_S comprend un exécutable de chargement à l'initialisation pour faire la mise à jour des champs avec JMRI. La portion de cet exécutable est protégée et ne peut pas être accidentellement effacée pendant un processus de mise à jour. Normalement les paramètres de l'utilisateur du QuadLN_S ne sont pas affectés par la mise à jour. Cependant il est conseillé de s'assurer que votre fichier de décodeur QuadLN_S est à jour avant de lancer la mise à jour ainsi vos paramètres peuvent être facilement rechargés.

Pour effectuer une mise à jour logicielle, suivez les étapes suivantes :

1. Téléchargez le nouveau logiciel depuis TamValleyDepot.com et placez dans votre répertoire préféré de JMRI.
2. Sélectionnez **Download Firmware** à partir du menu **LocoNet** de JMRI.
3. Cliquez sur le bouton format de fichier **16 bit** s'il n'est pas déjà sélectionné.
4. Cliquez sur le bouton **Select** et ensuite ouvrir le fichier à charger.
5. Cliquez sur **Read File** pour charger le fichier dans le téléchargeur.
6. Connectez la carte de configuration sur le Quad-LN.
7. Appuyez et maintenez en même temps les 3 boutons de la carte de configuration.
8. Après environ 2 secondes lorsque le voyant MSG s'assombrit, relâchez les boutons et la Led MSG devrait s'éteindre. Le QuadLN_S est maintenant en mode téléchargement.
9. Dans JMRI, cliquez sur **Download** pour commencer à télécharger le fichier dans le Quad-LN. La Led **MSG** clignote à chaque message reçu. La Led **ACTV** flache brièvement au fur et à mesure que la mémoire est écrite.
10. Si le téléchargement se termine avec succès, le QuadLN_S reprend un fonctionnement normal.
11. Si une erreur se produit, le voyant **ACTV** clignotera continuellement environ une fois par seconde. Le voyant MSG indique le code d'erreur.
 - Led MSG On = erreur de communication – mauvais message reçu.
 - Led MSG Off = erreur de configuration. Ceci peut être :
 - Une erreur de fichier de firmware – mauvais code constructeur ou mauvais code produit. Le fichier que vous voulez télécharger ne correspond pas à ce produit
 - Une erreur de version hardware – le logiciel n'est pas pour cette version de carte. Cherchez le bon logiciel pour la version hardware de votre carte.
 - Led MSG clignote une fois par seconde = erreur de version logiciel – le logiciel n'est pas plus récent que le logiciel existant. Si vous désirez toutefois charger cette version,

sélectionnez **Don't check software version** dans le téléchargeur de logiciel JMRI et répétez la procédure à l'étape 7.

Note : le logiciel du QuadLN_S est écrasé au fur et à mesure que le téléchargement s'effectue. Une fois que le téléchargement commence, s'il est interrompu ou si l'erreur 1 apparaît alors le logiciel du QuadLN_S sera probablement corrompu. Cependant la zone de lancement est intacte, donc cliquez juste sur **Abort**, coupez l'alimentation de la carte, appuyez et maintenez le bouton Select puis réappliquer l'alimentation à la carte. La LED **ACTV** doit être allumée et le voyant **MSG** doit être éteint. Maintenant, répétez le processus de téléchargement à partir de l'étape 9 ci-dessus.

Dans les cas où il y a beaucoup d'autres types de trafic LocoNet, essayez d'augmenter le Délai (ms) à la valeur de 100. Cela va ralentir le processus de téléchargement, mais il sera plus tolérant avec les collisions sur le LocoNet.

Si une erreur de configuration se produit, le QuadLN_S n'a pas accepté le téléchargement et le logiciel précédent est intact. Si vous voulez continuer à effectuer la mise à jour, de résoudre le problème alors répétez le processus de téléchargement.

7.2 MISE A JOUR DU MODELE DE DECODEUR

Si la mise à jour du firmware contient de nouvelles fonctionnalités, vous aurez besoin de la dernière version du fichier de modèle de décodeur JMRI pour profiter de ces fonctionnalités. La dernière version de JMRI doit contenir le dernier modèle du décodeur. Si vous n'utilisez pas la dernière version de JMRI et qu'une fonction de QuadLN_S semble manquer lorsque vous ouvrez une entrée de liste, suivez les étapes suivantes pour télécharger et installer les nouveaux fichiers de modèles de décodeur.

- Téléchargez les derniers modèles à partir du dossier /Decoder files/Tam Valley Depot/ dans la zone des fichiers de Yahoo JMRI Users Group.
- Copiez le fichier xml du décodeur dans le sous-dossier "decoders" à l'emplacement User File Location spécifié dans vos Préférences d'Emplacements de fichiers JMRI.
- Copiez le fichier xsl de décodeur dans le sous-dossier "xml/décodeurs" à l'emplacement du fichier de l'utilisateur spécifié dans vos JMRI Emplacements Préférences de fichier.
- Copiez le dossier de TVD à la "xml / décodeurs" sous-dossier de l'emplacement du fichier de l'utilisateur spécifié dans vos Emplacements de Préférences de fichier JMRI.
- Sélectionnez **Recreate Decoder Index** dans le menu **Debug** JMRI.

8 TABLEAU DES CV

8.1 TABLEAU COMPLET DES CVS

CV #	Description	Etendue	Valeur par défaut
1	Réservé		
3,2	première adresse de servo	0-2044	0 (0,0)
4	Version de logiciel	0-255	-
5	Numéro de logiciel	0-255	-
6	Version hardware	0-255	-
7	ID du produit	6	6
8	ID du constructeur	59	59
10,9	Adresse de début des IO Main	0-4092	0-4092
11	Configuration	Tableau 8-3	180
12	Configuration 2	Tableau 8-4	0
14,13	Adresse de début des verrouillages	0-2044	101 (0,101)
16,15	Adresse de début des IO Aux	0-4092	4 (0,4)
17,18	Adresse longue	Tableau 8-2	11000 (234,248)
19-58	Réservé		
59-250	Aspect de signal 1-48	Tableau 8-14	
251-256	Réservé		
257-264	Servo 1-8 vitesse	Tableau 8-5	4
265-272	Servo 1-8 vitesse Dévié	Tableau 8-5	4
273-288	Servo 1-8 Position Fermé	Tableau 8-6	1260 (4,236)
289-304	Servo 1-8 Position Dévié	0-2400	1140 (4,116)
305-320	Servo 1-8 adresse de cascade	Tableau 8-7	0 (0,0)
321-324	Option des IO Aux 1-4	Tableau 8-8	13
325-328	Option des IO Main 1-4	Tableau 8-8	13
329-332	Détecteur des IO Main 1-4	Tableau 8-9	16
333-340	IO Aux 1-4 secondaire	Tableau 8-10	0 (0,0)
341-348	IO Main 1-4 secondaire	Tableau 8-10	0 (0,0)
349-364	Action des IO Aux 1-4	Tableau 8-11	Tableau 8-11
365-380	Action des IO Main 1-4	Tableau 8-11	Tableau 8-11
381-512	Réservé		
513-536	Lampes de signal 1-24	Tableau 8-18	160
537-632	Adresse de signal 1-48	Tableau 8-16 Tableau 8-17	0
633-656	Led de signal 1-24	Tableau 8-19	159
657-704	Configuration d'aspect de signal 1-48	Tableau 8-15	0
705-768	Réservé		
769-776	Entrée de la route 1 1-4	Tableau 8-12	0 (0,0)
777-784	Entrée de la route 2 1-4	Tableau 8-12	0 (0,0)
785-792	Entrée de la route 3 1-4	Tableau 8-12	0 (0,0)
793-800	Entrée de la route 4 1-4	Tableau 8-12	0 (0,0)
801-808	Entrée de la route 5 1-4	Tableau 8-12	0 (0,0)
809-816	Entrée de la route 6 1-4	Tableau 8-12	0 (0,0)
817-824	Entrée de la route 7 1-4	Tableau 8-12	0 (0,0)
825-832	Entrée de la route 8 1-4	Tableau 8-12	0 (0,0)
833-896	Réservé		
897-904	Contrôle de route 1-8	Tableau 8-13	0
905-	Réservé		

Pour les entrées d'adresse, le numéro de l'équipement LocoNet associé est (adresse + 1).

8.2 FORMAT DE CV ADRESSE LONGUE

CV	17								18							
Bit	1	1	h	h	h	h	h	h	l	l	l	l	l	l	l	l
Valeur par Défaut	1	1	1	0	1	0	1	0	1	1	1	1	1	0	0	0

h = bits d'adresse haute

l = bits d'adresse basse

8.3 FORMAT DES CV DE CONFIGURATION

CV									
Bit	b	Lb	l	o	si	se	e	e	
Valeur par Défaut	1	0	1	1	0	1	0	0	

b = Busy Retry (réessayer sur occupation du LocoNet) : 0 inactif, 1 actif

lb = Lock Blink state (état de clignotement du verrouillage) : 0 verrouillé, 1 déverrouillé

l = Local Action (action locale) : 0 interne, 1 visible sur le LocoNet

o = Output Off switch request (requête de commutateur d'état de la sortie) : 0 ne pas envoyer, 1 envoyer

si = Signal State memory (mémoire d'état du signal) : 0 = Off, 1 = On

se = Servo State Memory (mémoire d'état de servo) : 0 = Off, 1 = On

e = Port d'extension : 0 = commande de Relais, 1 = commande de signal, 2 = commande de Servo (mose Octo), 3 = réservé.

8.4 FORMAT DE CV DE LA CONFIGURATION 2

CV								
Bit	r	r	Rd	Rd	R	C	D	P
Valeur par Défaut	0	0	0	0	0	0	0	0

r = Réserve

rd = délai de la route : 0 = 1 sec, 1 = 2 sec, 2 = 4 sec, 3 = 8 sec

c = pas de station de commande : 0 = inactif, 1 = actif

d = pilote du servo : 0 = toujours actif, 1 = inactif quand il est stoppé

p = inversion d'entrée : 0 = normal, 1 = inversé

8.5 FORMAT DE CV DE LA VITESSE DE SERVO

CV									
Bit	r	R	S	S	S	S	S	S	
Valeur par Défaut	0	0	0	0	0	1	0	0	

R = mode RapidStart : 0=non, 1= 1/8, 2 = ¼, 3 = 3/8

s = speed : 0 = la plus basse, 63 = la plus rapide

8.6 FORMAT DE CV DE POSITION FERMEE DU SERVO

CV																
Bit	D	L	L	M	h	h	h	h	l	l	l	l	l	l	l	l
Valeur par Défaut	0	0	0	0	0	1	0	0	1	1	1	0	1	0	0	0

D = vitesse directionnelle : 0 = inactif, 1 = actif

L = mode de verrouillage : 0 = non, 1 = verrouillage local, 2 = milieu, 3 = réservé

M = message : 0 = non, 1 = report exact

H = bits de position hauts

l = bits de position bas

8.7 FORMAT DE CV DES ADRESSES EN CASCADE DE SERVO

CV																
Bit	T	T	A	A	X	H	H	H	L	L	L	L	L	L	L	L
Valeur par Défaut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

T = déclencheur : 0 = non, 1 = fermé, =2 = dévié, 3 = fermé ou dévié

X = non utilisé

A = Action : 0 fermé, 1 dévié, 2 suivant, 3 inversé

H = bits d'adresse haute

L = bits d'adresse basse

8.8 FORMAT DE CV DE L'OPTION IO

CV								
Bit	F	Ty	S	L	L	L	Tr	Tr
Valeur par Défaut	0	0	0	0	1	1	0	1

F = DCC Freeze (gel du DCC) : 0 toujours actif, 1 en cas de perte du DCC

Ty = type d'entrée : 0 = générale, 1 = détecteur

S = Led Sense (sens de la Led) : 0 = inactif, 1 = fixe, 2 = clignote pendant de le mouvement, 3 = clignote pendant de mouvement ou le verrouillage, 4 = clignote pendant le verrouillage, 5 = état du verrouillage, 6 = position, 7 = clignote à dévié.

Tr = Trigger (déclenchement) 0 = non, 1 = front positif, 2 = front négatif, 3 = les 2 fronts

8.9 FORMAT DE CV DE L'OPTION DETECTEUR SUR IO MAIN – ENTREE DETECTEUR

CV								
Bit	T	R	R	S	S	S	S	S
Valeur par Défaut	0	0	0	1	0	0	0	0

T = type d'entrée : 0 = consommation de courant, 1 = Infrarouge

R = réservé

S = sensibilité : 0-31

8.10 FORMAT DE CV DE L'ADRESSE SECONDAIRE IO MAIN

CV																
Bit	T	D	C	C	H	H	H	H	L	L	L	L	L	L	L	L
Valeur par Défaut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

T = Type : 0 = déclencher sur ce message, 1 = envoyer ce message

D = Device (équipement) : 0 = aiguillage, 1 = capteur

C = Condition : 0 = none, 1 = Hi/Closed, 2 = Lo/Thrown, 3 = les deux/suivant

H = bits d'adresse hauts

L = bits d'adresse bas

8.11 FORMAT DE CV DE L'ACTION D'IO

CV																
Bit	H	H	H	H	G	G	G	G	F	F	F	F	E	E	E	E
Valeur par Défaut	H	H	H	H	G	G	G	G	F	F	F	F	E	E	E	E
CV																
Bit	D	D	D	D	C	C	C	C	B	B	B	B	A	A	A	A
Valeur par Défaut	D	D	D	D	C	C	C	C	B	B	B	B	A	A	A	A

H = Servo 8 (seulement en mode Octo)

G = servo 7 (seulement en mode Octo)

F = servo 6 (seulement en mode Octo)

E = servo 5 (seulement en mode Octo)

D = servo 4

C = servo 3

B = servo 2

A = servo 1

0 = non, 1 = bascule, 2 = suivant, 3 = inverse, 4 = fermé, 5 = dévié, 9 = bascule verrouillé, 10 = suivant verrouillé, 11 = inverse verrouillé, 12 = fermé verrouillé, 13 = dévié verrouillé

Par défaut : Pour le servo dans le même groupe, la valeur par défaut est 1. Pour les autres servos, la valeur par défaut est 0.

8.12 FORMAT DE CV DES ENTREES DE ROUTES

CV	paires								impairs							
Bit	R	A	A	A	H	H	H	H	L	L	L	L	L	L	L	L
Valeur par Défaut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

R = réservé

A = action : 0 = non, 1 = aiguillage fermé, 2 = aiguillage dévié, 3 = capteur niveau haut, 4 = capteur niveau bas

H = bits d'adresse hauts

L = bits d'adresse bas

8.13 FORMAT DE CV DE CONTROLE DE ROUTES

CV								
Bit	R	R	R	R	R	R	T	T
Valeur par Défaut	0	0	0	0	0	0	0	0

R = réservé

T = type : 0 = non, 1 = basique, 2 = sélecteur, 3 = étendu

Le type = 3 n'est pas utilisé sur la route 1

8.14 FORMAT DE CV D'ASPECT DE SIGNAL

CV									
Bit	B	M	M	P	P	P	P	p	
Valeur par Défaut	0	0	0	0	0	0	0	0	

B = bicolore : 0 = non, 1 = oui

Mm = Mode : 0 = non, 1 = constant, 2 = clignotement phase A, 3 = clignotement phase B

Ppppp = broche : 0-31

8.15 FORMAT DE CV DE CONFIGURATION D'ASPECT DE SIGNAL

CV								
Bit	I	M	R	R	R	R	R	R
Valeur par Défaut	0	0	0	0	0	0	0	0

I = en cours d'utilisation : 0 = non, 1 = oui

A = mode d'adressage : 0 = mode signal, 1 = mode aiguillage

R = réservé

8.16 FORMAT DE CV D'ADRESSAGE D'ASPECT DE SIGNAL – MODE SIGNAL

CV															
Bit	A	A	A	A	A	H	H	H	L	L	L	L	L	L	L
Valeur par Défaut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

A = aspect: 0-31

H = bits d'adresse hauts du mat

L = bits d'adresse bas du mat

8.17 FORMAT DE CV D'ADRESSAGE D'ASPECT DE SIGNAL – MODE AIGUILLAGE

CV															
Bit	M	M	M	M	S	H	H	H	L	L	L	L	L	L	L
Valeur par Défaut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

M = Mat : 0-15

S = ensemble de l'aiguillage : 0 = dévié, 1 = fermé

H = bits d'adresse hauts d'aiguillage

L = bits d'adresse bas d'aiguillage

8.18 FORMAT DE CV DE SIGNAL A LAMPE

CV								
Bit	F	Fr	T	R	R	R	R	R
Valeur par Défaut	1	0	1	0	0	0	0	0

R = réservé

T = type : 0 = pas de projecteur, 1 = projecteur

Fr = taux de clignotement : 0 = signal, 1 = passage à niveau

F = fondu : 0 = non, 1 = oui

8.19 FORMAT DE CV DE SIGNAL A LED

CV								
Bit	G	E	E	B	B	B	B	B
Valeur par Défaut	1	0	1	0	0	0	0	0

G = groupe de départ : 0 = non, 1 = oui

E = effet : 0 = Off, 1 = On, 2 = bicolore, 3 = réservé

B = luminosité : 0-31

9 NOTES DU TRADUCTEUR

Lors de la première programmation du QuadLN_S par JMRI, j'ai quand même tâtonné. Donc j'apporte ici mon expérience. Cette procédure a été réalisée après avoir chargé JMRI 3.8 en Français depuis le site de Boutique Train en mettant à jour le driver suivant

http://www.tamvalleydepot.com/images/QuadLN_7_18_12.zip dans le répertoire C:\Program Files (x86)\JMRI\xml\decoders . Il est à jour dans la version 3.9 non francisée.

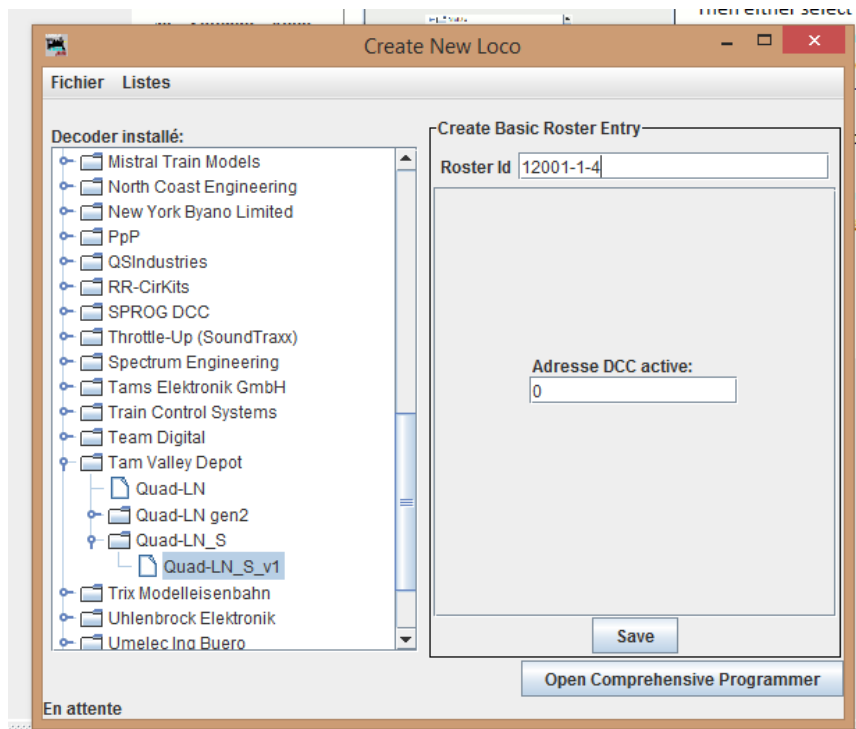
9.1 CONNEXION

Le QuadLN_S ne possédant que des ports de communication LocoNet, il faut le brancher à une console supportant le LocoNet. Généralement on le branche à un DCS 100 ou 200 pour obtenir un LocoNet complet avec tous les signaux et surtout le RailSync qui est l'horloge. Ensuite on peut se servir d'un LocoBuffer-USB branché dans un des 2 ports LocoNet pour relier le PC avec JMRI, le QuadLN_S étant branché sur l'autre entrée du DCS. Le LocoBuffer-USB ne fournit pas seul le RailSync (il faut l'ouvrir et le strapper pour qu'il puisse fonctionner seul). Ce montage sans d'autres équipements est nécessaire pour la première mise service du QuadLN_S pour lui changer son adresse longue fixée d'usine à 11000.

9.2 DECLARATION

Une fois tout branché, dans DecoderPro3, placez-vous en mode Voie de programmation (Programming Track) en bas à droite de la page d'accueil.

Cliquer en haut à gauche sur « + Locomotive », une nouvelle fenêtre s'ouvre, puis sélectionner le QuadLN_S



Entrer un nom dans Roster Id. je prends toujours en premier l'adresse qui sera affectée au décodeur pour éviter les doublons et une lecture directe des adresses utilisées. Puis cliquez sur « save ».

Puis cliquer sur « Open Comprehensive Programmer » et la fenêtre de programmation s'ouvre.

Entrez les informations qui vous conviennent pour identifier rapidement à quoi servira le décodeur
Puis cliquez sur enregistrer dans la liste, il apparait dans la liste de la page principale.

9.3 PROGRAMMATION DE L'ADRESSE DE LA CARTE

Cliquez sur l'onglet Basic et entrer l'adresse longue qui vous permettra par la suite de programmer et modifier la programmation en ligne sans isoler chaque décodeur (plage d'adresses de 1 à 16383)

Après avoir mis le strap ADDR sur la carte, les 2 voyants Actv et Msg clignotent alternativement.

Cliquez sur « écrire les modifications de cet onglet » et « sauvegardez » dans le menu fichier. Fermer cette fenêtre ainsi que la précédente pour ne plus avoir d'autres fenêtres ouvertes que celle de la liste principales. Enlever le strap ADDR.

Passez en mode programmation sur la voie principale, en bas à droite

9.4 PROGRAMMATION DES ADRESSES DE BASES DES ACCESSOIRES

Sélectionnez votre décodeur dans la liste, puis appuyez sur le bouton Program, pour continuer à programmer votre décodeur.

Dans l'onglet « QuadLN_S », entrer les éléments suivants :

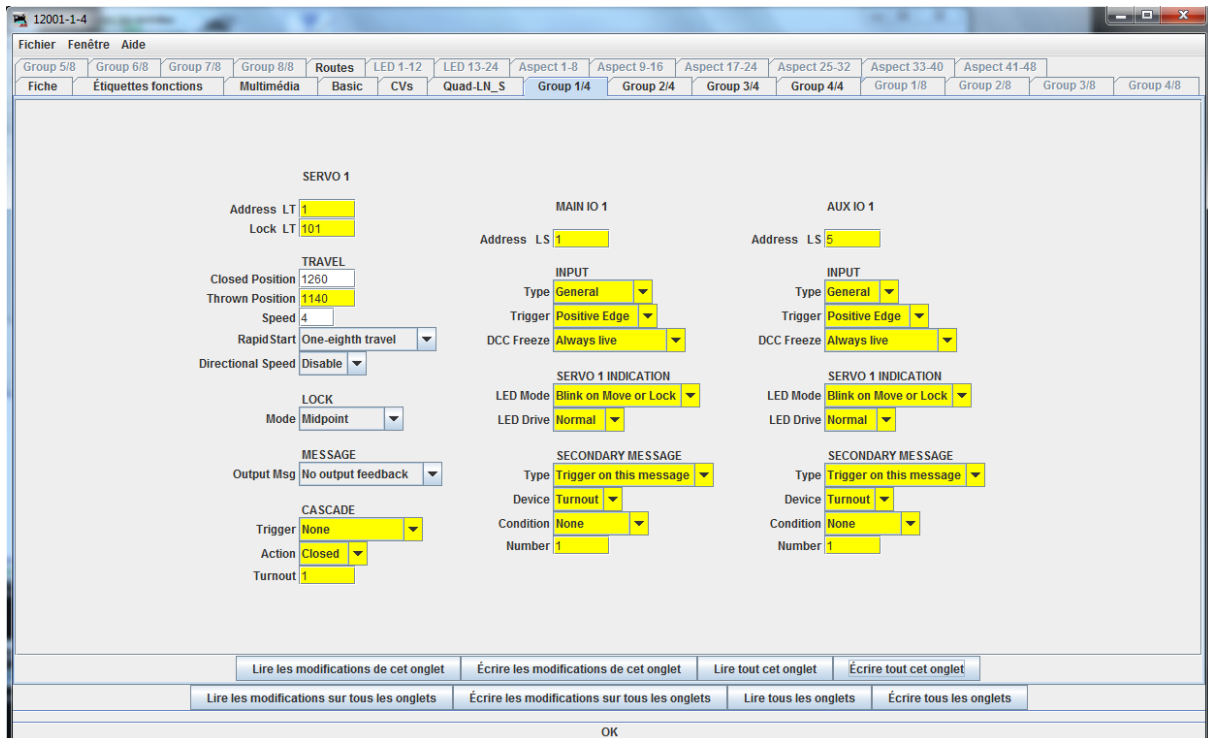
1. Servo Start Address : entrez la première adresse pour 1 groupe de 8 servos. Le QuadLN_S peut être utilisé avec un mode 4 ou 8 servos.
2. Lock Start Address : entrez la première adresse pour 1 groupe de 8. Ces adresses ne doivent pas chevaucher les adresses des servos.
3. Main IO Start Address : entrez la même adresse que pour la première adresse des servos.
4. Aux IO Start Address : entrez la même adresse que pour la première adresse des servos + 4.
5. Sélectionnez le mode de fonctionnement
 - a. Relays 1-4 : 4 servos + 4 relais (aiguillages Electrofrog)
 - b. Servos 5-8 : 8 servos (aiguillages Insulfrog)
 - c. Signaling : 4 servos + 24 signaux (aiguillages Insulfrog)
6. Servo OFF When Stopped : les servos ne sont plus alimentés à l'arrêt, à n'utiliser que si vos aiguillages ont un ressort de point milieu qui maintient les aiguilles en position. Généralement sur les aiguillages Peco de ce type on enlève le ressort afin d'avoir un vrai mouvement lent. Ne pas cocher cette case pour maintenir les aiguilles en position.

7. Servo State Memory : cochez cette fonction, en cas de coupure d'alimentation et de son rétablissement ou de verrouillage en position milieu, les servos reprennent leurs dernières positions connues.
8. Signal State Memory : cocher cette case si vous vous servez des entrées/sorties Main et Aux ce qui permettra dans le cas de coupure d'alimentation et de son rétablissement, de mémoriser l'état des indicateurs que vous auriez positionnés.
9. Retry On DCC Busy : en général on décoche cette case, sauf au cas où le QuadLN_S doit envoyer des informations vers le LocoNet. Ce cas de figure est relativement rare et se présente quand il adresse d'autres décodeurs dans le cas de routes complexes où les aiguillages sont pilotés par d'autres cartes
10. Output Off Switch Request : décochez cette case car elle augmente le trafic LocoNet et ne sert que dans certains cas spécifiques (chapitre 6.7)
11. No Command Station : au cas où il n'y ait pas de station de commande. Mais il faut quand même un RailSync pour le LocoNet...
12. Input invert : ne pas toucher. A voir en fonction de l'utilisation des entrées.
13. Blink Indicates Unlock : ne pas toucher. Voir le chapitre 6.10.
14. Broadcast Local Actions : Ce paramètre détermine si le QuadLN_S rend visible sur le LocoNet les actions locales de servo et de verrouillage. La valeur par défaut est cochée ce qui signifie que les Actions locales sont diffusées. Vous pouvez le décocher si vous n'avez pas à programmer le décodeur, une fois dévalidé vous ne pouvez plus lire les informations dans le décodeur, vous devez le revalider pour lire à nouveau.

Cliquez sur « écrire les modifications de cet onglet » et « sauvegardez » dans le menu fichier.

9.5 PROGRAMMATION DE CHAQUE ACCESSOIRE

Puis passez sur chaque onglet Group 1/4 à 4/8 dans le cas de l'option Relays 1-4 ou sur Group 1/8 à 8/8 dans le cas de l'option Servos 5-8 pour régler les aiguillages en fonction des positions que vous utilisez (Servos 1-4 et/ou expansion 1-4).

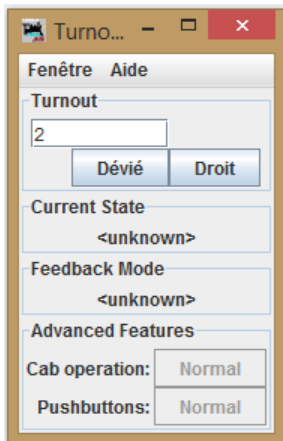


1. L'adresse LT correspond à l'adresse fixée de première adresse de servo. (1 pour group 1, 2 pour group 2, etc.)
2. Le Lock LT correspond à l'adresse fixée de première adresse de Lock. (101 pour group 1, 102 pour group 2, etc.)
3. Il faut jouer sur Closed et Thrown position pour régler la position de l'aiguille sur les 2 positions dévié et droit. Réglage possible de 0 à 2400
4. Speed peut être réglée pour avoir un effet plus réaliste en la descendant à 2. Mais il serait conseillé de régler le Rapid Start pour un démarrage du mouvement plus violent et vaincre le frottement plus important en partant d'une vitesse nulle.
5. Rapid Start à 1/8 de la course ($1260-1140=120/8=15$)
6. Directionnall Speed Disabled , à activer si on veut des vitesses différentes dans chaque sens de mouvement.
7. Lock Mode Midpoint : avec une commande Lock le servo se met à sa position centrale. On peut jouer également pour la fixation des servos en position médiane sur Closed et Thrown position en les mettant à 1200 par exemple, et une fois le servo fixé sous tension effectuer le réglage des 2 courses.
8. Output Msg est paramétré à Exact Feedback si le logiciel d'exploitation du réseau (TrainControler) est capable de prendre en compte l'information
9. Les autres paramètres liés à des routes et aux entrées sorties ne sont pas nécessaires dans notre cas car nous n'utilisons pas de TCO manuels.

Cliquez sur « écrire les modifications de cet onglet » et « sauvegardez » dans le menu fichier.

9.6 TESTS

Fermez la fenêtre puis testez en allant dans le menu Actions de la fenêtre et en cliquant sur Turnout Control



Cliquez sur Dévié et Droit pour vérifier le fonctionnement de l'aiguillage. Si les commandes sont inversées, ce n'est pas grave. Il faudra vérifier dans TrainControler si ces commandes sont inversées par rapport au basculement de l'aiguillage à l'écran, puis agir dans les propriétés dans TrainControler.