

# TeSys U LUFP9

## Passerelle DeviceNet / Modbus RTU Guide d'exploitation

03/2009



---

Schneider Electric ne saurait être tenu responsable des erreurs pouvant figurer dans le présent document. Si vous avez des suggestions, des améliorations ou des corrections à apporter à cette publication, veuillez nous en informer.

Aucune partie de ce document ne peut être reproduite sous quelque forme que ce soit, ni par aucun moyen que ce soit, électronique ou mécanique, y compris la photocopie, sans la permission écrite expresse de Schneider Electric.

Toutes les réglementations de sécurité locales pertinentes doivent être observées lors de l'installation et de l'utilisation de ce produit. Pour des raisons de sécurité et afin de garantir la conformité aux données système documentées, seul le fabricant est habilité à effectuer des réparations sur les composants.

Lorsque des équipements sont utilisés pour des applications présentant des exigences de sécurité techniques, suivez les instructions appropriées.

La non-utilisation du logiciel Schneider Electric ou d'un logiciel approuvé avec nos produits peut entraîner des blessures, des dommages ou un fonctionnement incorrect.

Le non-respect de cette consigne peut entraîner des lésions corporelles ou des dommages matériels.

© 2009 Schneider Electric. Tous droits réservés.

# Table des matières

<b>Table des matières .....</b>	<b>3</b>
<b>Informations de sécurité .....</b>	<b>4</b>
<b>A propos de ce guide .....</b>	<b>5</b>
<b>1. Introduction .....</b>	<b>6</b>
1.1. Introduction du Guide d'exploitation .....	6
1.2. Présentation de la passerelle LUF9 .....	8
1.3. Terminologie .....	8
1.4. Présentation de l'architecture « système » des communications .....	9
1.5. Principe de configuration et de fonctionnement de la passerelle .....	10
<b>2. Mise en œuvre matérielle de la passerelle LUF9 .....</b>	<b>12</b>
2.1. Réception .....	12
2.2. Présentation de la passerelle LUF9 .....	12
2.3. Montage de la passerelle sur rail DIN .....	13
2.4. Alimentation de la passerelle .....	14
2.5. Raccordement de la passerelle au réseau Modbus .....	14
2.5.1. Exemples de topologies de raccordement Modbus .....	15
2.5.2. Brochage .....	17
2.5.3. Recommandations de câblage du réseau Modbus .....	18
2.6. Connexion de la passerelle LUF9 au réseau DeviceNet .....	20
2.7. Configuration des fonctions de communication DeviceNet .....	21
2.7.1. Codage de la vitesse DeviceNet .....	21
2.7.2. Codage de l'adresse de la passerelle .....	22
2.7.3. Exemples de configuration de la passerelle .....	22
<b>3. Signalisation .....</b>	<b>23</b>
<b>4. Mise en œuvre logicielle de la passerelle .....</b>	<b>25</b>
4.1. Introduction .....	25
4.1.1. Architecture système .....	25
4.1.2. Configuration des départs-moteurs .....	26
4.1.3. Temps de cycle Modbus .....	26
4.1.4. Gestion des modes dégradés avec la configuration par défaut de la passerelle .....	26
4.2. Configuration de la passerelle sous RSNetWorx .....	31
4.2.1. Sélection et ajout du scanner DeviceNet de l'automate maître .....	31
4.2.2. Mise en place du fichier de description de la passerelle .....	31
4.2.3. Sélection et ajout d'une passerelle au réseau DeviceNet .....	32
4.2.4. Modification des paramètres de la passerelle .....	32
4.2.5. Configuration du Scanner DeviceNet .....	34
4.2.6. Configuration des entrées issues de la passerelle .....	35
4.2.7. Configuration des sorties destinées à la passerelle .....	36
4.2.8. Transfert de la configuration du scanner DeviceNet .....	36
4.2.9. Développement d'une application DeviceNet .....	37
4.3. Description des services affectés aux entrées/sorties de la passerelle .....	37
<b>5. Initialisation et diagnostic de la passerelle .....</b>	<b>39</b>
5.1. Gestion complète .....	39
5.1.1. Mot de commande du maître DeviceNet .....	39
5.1.2. Mot d'état de la passerelle .....	40
5.2. Diagnostic et commande .....	40
5.2.1. Mot de commande du maître DeviceNet .....	40
5.2.2. Mot d'état de la passerelle .....	41
5.3. Fonctionnement simplifié .....	41
5.4. Description du mot de commande du maître DeviceNet .....	42
5.5. Description du mot d'état de la passerelle .....	45
<b>6. Configuration de la passerelle .....</b>	<b>48</b>
6.1. Raccordement de la passerelle au PC de configuration .....	48
6.1.1. Brochage .....	49
6.1.2. Protocole de la liaison RS-232 .....	49
6.2. Installation de ABC-LUF9 Config Tool .....	50
6.3. Connexion / déconnexion de la passerelle .....	50
6.4. Importation de la configuration de la passerelle .....	51
6.5. Transfert d'une configuration vers la passerelle .....	52
6.6. Suivi du contenu de la mémoire de la passerelle .....	52
6.7. Suppression d'un esclave Modbus .....	55
6.8. Ajout d'un esclave Modbus .....	56
6.9. Modification des données périodiques échangées avec un esclave Modbus .....	58
6.9.1. Remplacement d'une donnée périodique d'entrée .....	58
6.9.2. Remplacement d'une donnée périodique de sortie .....	59
6.9.3. Augmentation du nombre des données périodiques d'entrée .....	60
6.9.4. Augmentation du nombre des données périodiques de sortie .....	64
6.10. Suppression des données aperiodiques de paramétrage .....	69
6.11. Modification de la configuration d'un esclave Modbus .....	71
6.11.1. Modification du nom d'un esclave Modbus .....	72
6.11.2. Modification de l'adresse d'un esclave Modbus .....	72
6.11.3. Modification du nom d'une commande ou d'une transaction Modbus .....	73
6.12. Ajout et paramétrage d'une commande Modbus .....	74
6.12.1. Cas des départs-moteurs TeSys U .....	74
6.12.2. Cas d'un esclave Modbus générique .....	76
6.12.3. Ajout d'une commande Modbus spéciale .....	90
6.13. Configuration des caractéristiques générales de la passerelle .....	97
6.13.1. Élément « Fieldbus » .....	97
6.13.2. Élément « ABC-LUF9 » .....	98
6.13.3. Élément « Sub-Network » .....	101
6.14. Ajout d'un nœud de diffusion .....	103
<b>Annexe A : Caractéristiques techniques .....</b>	<b>104</b>
<b>Annexe B : Configuration par défaut .....</b>	<b>107</b>
<b>Annexe C : Exemple d'utilisation (RSLogix 500) .....</b>	<b>111</b>
<b>Annexe D : Objets DeviceNet .....</b>	<b>120</b>
<b>Annexe E : Commandes Modbus .....</b>	<b>140</b>
<b>Index .....</b>	<b>144</b>
<b>Glossaire .....</b>	<b>145</b>

# Informations de sécurité

## AVIS

Lisez attentivement ces consignes et examinez l'appareil afin de vous familiariser avec l'équipement avant de l'installer, de l'utiliser ou d'en assurer la maintenance. Vous pourrez voir apparaître les messages spéciaux suivants tout au long de cette documentation ou sur l'appareil. Ils ont pour but de vous mettre en garde contre des risques potentiels ou d'attirer votre attention sur des informations qui clarifient ou simplifient une procédure.



L'apposition de ce symbole à un panneau de sécurité Danger ou Avertissement signale un risque électrique pouvant entraîner des lésions corporelles en cas de non-respect des consignes.



Ceci est le symbole d'une alerte de sécurité. Il vous avertit d'un risque de blessures corporelles. Respectez scrupuleusement les consignes de sécurité associées à ce symbole pour éviter de vous blesser ou de mettre votre vie en danger.

## DANGER

**DANGER** indique une situation immédiatement dangereuse qui, si elle n'est pas évitée, **entraînera** la mort ou des blessures graves.

## AVERTISSEMENT

L'indication **AVERTISSEMENT** signale une situation potentiellement dangereuse et susceptible **d'entraîner la mort** ou des blessures graves.

## ATTENTION

L'indication **ATTENTION** signale une situation potentiellement dangereuse et susceptible **d'entraîner des** blessures d'ampleur mineure à modérée.

## ATTENTION

L'indication **ATTENTION**, utilisée avec le symbole d'alerte de sécurité, signale une situation potentiellement dangereuse et susceptible **d'entraîner des** dommages aux équipements.

## Remarque importante

L'installation, l'utilisation, la réparation et la maintenance des équipements électriques doivent être assurées par du personnel qualifié uniquement. Schneider Electric décline toute responsabilité quant aux conséquences de l'utilisation de cet appareil. Une personne qualifiée est une personne disposant de compétences et de connaissances dans le domaine de la construction et du fonctionnement des équipements électriques et installations et ayant bénéficié d'une formation de sécurité afin de reconnaître et d'éviter les risques encourus.

# A propos de ce guide

---

## Remarque concernant la validité

Ce document s'applique à toutes les passerelles V2.

Fonctions et améliorations par rapport aux précédentes versions du produit :

- Augmentation du nombre d'instances/transactions de 50 à environ 100.
- Protection par mot de passe du transfert/chargement de la configuration dans LUF9.
- Fonction de débogage avec un analyseur de ligne (Sub-network Line Analyzer).
- Amélioration du comportement du déclenchement d'une réponse (response trigger).
- Possibilité d'association de fichiers de configuration MS Windows (\*.CFG). Ouverture automatique du fichier de configuration, sur double clic, dans ABC-LUF9 Config Tool.
- Fonctionnalité d'affichage étendu en surveillance de noeud (différentes largeurs de colonne et affichage hexadécimal/décimal).
- Utilisation simplifiée. Nouveautés et améliorations du menu d'options.

Les données et illustrations fournies dans ce guide ne sont pas de nature contractuelle. Nous nous réservons le droit de modifier nos produits conformément à notre politique de développement permanent. Les informations figurant dans ce document peuvent faire l'objet de modifications sans préavis et ne doivent pas être interprétées comme un engagement de la part de Schneider Electric.

---

## Documents associés

Titre du document	Référence
AnyBus Communicator – User Manual	ABC_User_Manual.pdf (SDN-7061-059)
Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control	NEMA ICS 1.1 (nouvelle édition)
Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems	NEMA ICS 7.1 (nouvelle édition)
Modbus User Guide	TSX DG MDB E
Modicon Modbus Protocol Reference Guide	PI-MBUS-300 Rev. J

Vous pouvez télécharger ces publications et autres informations techniques depuis notre site web à l'adresse : [www.schneider-electric.com](http://www.schneider-electric.com).

---

## Commentaires des utilisateurs

Envoyez vos commentaires à l'adresse e-mail [techcomm@schneider-electric.com](mailto:techcomm@schneider-electric.com).

# 1. Introduction

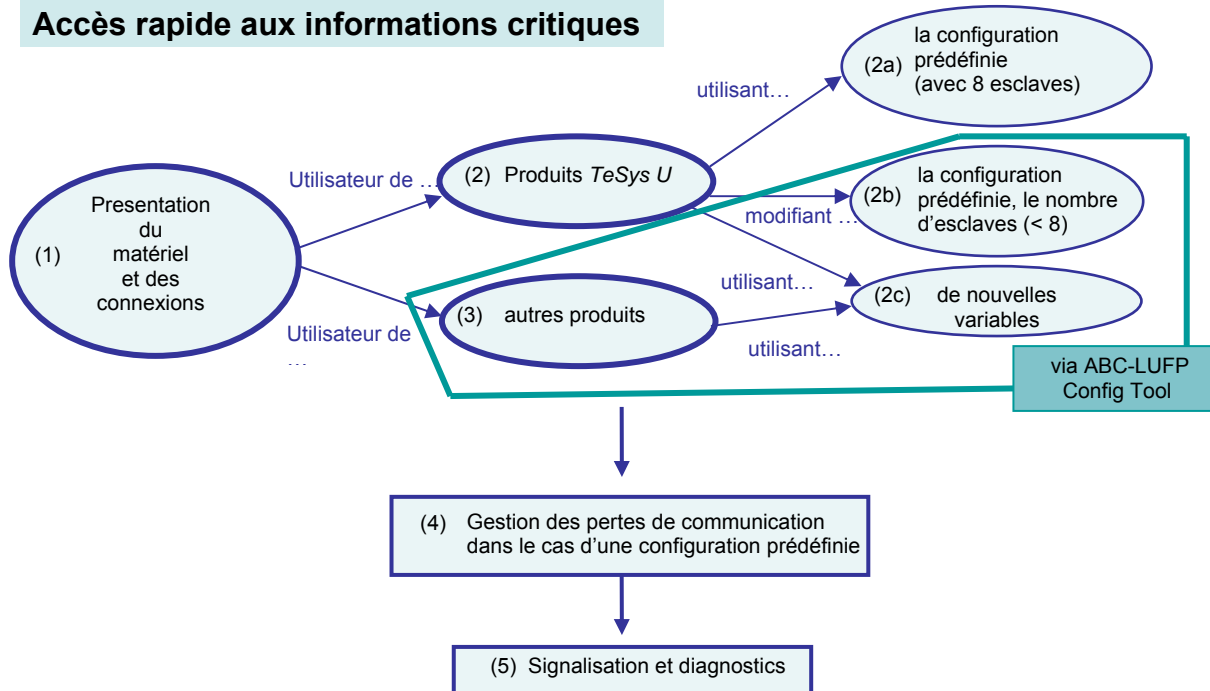
---

## 1.1. Introduction du Guide d'exploitation

- Chapitre 1** Ce chapitre décrit la passerelle, son guide d'exploitation ainsi que les termes qui y sont employés.
- Chapitre 2** Ce chapitre présente la passerelle et décrit l'ensemble des éléments à manipuler lors de sa mise en œuvre, qu'ils soient internes (roues codeuses) ou externes (câbles et connecteurs) à la passerelle.
- Chapitre 3** Ce chapitre décrit les six DEL situées sur la face avant de la passerelle.
- Chapitre 4** Ce chapitre décrit les étapes successives permettant de mettre en œuvre la passerelle dans sa configuration par défaut, avec un automate utilisant DeviceNet. Les passerelles LUF9 sont livrées pré-configurées pour permettre d'interfacer un maître DeviceNet avec 8 esclaves Modbus prédéfinis (départs-moteurs TeSys U).
- Chapitre 5** Ce chapitre décrit deux registres présents dans la mémoire de la passerelle, ceux-ci étant réservés à l'initialisation et aux diagnostics de la passerelle. Ils sont uniquement échangés entre le maître DeviceNet et la passerelle.
- Chapitre 6** Ce chapitre décrit l'utilisation du logiciel « ABC-LUF9 Config Tool », qui permet de modifier ou de créer une nouvelle configuration destinée à la passerelle, et présente les différentes fonctions de ce logiciel (ajout ou suppression d'un esclave Modbus, ajout ou modification d'une commande Modbus, etc.).
- Ce chapitre présente également les changements à reporter sur les opérations de mise en œuvre logicielle sous RSNetWorx.
- Annexe A** Cette annexe décrit les aspects techniques de la passerelle et des réseaux auxquels elle est interfacée, c'est-à-dire les réseaux DeviceNet et Modbus RTU.
- Annexe B** Cette annexe décrit les principales caractéristiques de la configuration par défaut de la passerelle LUF9, sans toutefois rentrer dans les détails liés à ABC-LUF9 Config Tool.
- Annexe C** Cette annexe décrit un exemple simple d'utilisation de la configuration par défaut de la passerelle LUF9. Cet exemple exploite les registres de commande et de surveillance de 8 départs-moteurs TeSys U et utilise les services aperiodiques de lecture et d'écriture de la valeur d'un paramètre de départ-moteur.
- Annexe D** Cette annexe décrit les objets DeviceNet génériques ainsi que les objets DeviceNet spécifiques à la passerelle LUF9. Les valeurs des attributs de ces objets y sont également fournies.
- Annexe E** Cette annexe décrit le contenu des trames des commandes Modbus supportées par la passerelle LUF9.

# 1. Introduction

## Accès rapide aux informations critiques



### (1) Présentation du matériel et des connexions

- **Voir Chapitre 2**
- alimentation,
  - installation,
  - raccordement au réseau Modbus,
  - raccordement au réseau DeviceNet,
  - sélection de la vitesse de transmission et des adresses

### (2) Utilisateur de produits TeSys U

#### (2a) avec 8 esclaves

- **Voir Chapitre 4**

#### (2b) réduction du nombre d'esclaves

- **Voir Chapitre 6**
- Utilisation de ABC-LUFP Config Tool :
- installation (6.2),
  - connexion (6.1),
  - retrait d'esclaves (6.6)

#### (2c) accès à de nouvelles variables

- **Voir Chapitre 6**
- Utilisation de ABC-LUFP Config Tool pour accéder à des registres autres que les registres standard 704 (Command) et 455 (Status)
- avec la même requête :
- remplacement d'un registre par un autre (par exemple 455 par 458)
  - augmentation de la taille (le nombre de registres)
- avec une requête additionnelle :
- ajout de commandes supplémentaires
  - autres opérations (6.7 à 6.11)

### (3) Utilisateur d'autres produits génériques Modbus

- **Voir Chapitre 6 (6.7 à 6.11, 6.11.2)**
- Choisissez entre :
- adapter la configuration prédéfinie fournie avec la passerelle, si elle est suffisamment proche de celle souhaitée (1 registre de lecture et 1 registre d'écriture, 1 adresse de registre à modifier), ou
  - définir votre propre configuration (voir le manuel d'utilisation de ABC)

### (4) Perte de communication

- **Voir les chapitres 4.1.4.1 et 6.11.2.2**
- Les variables décrites sont :
- **Reconnect time** (unité = 10ms, valeur par défaut = 10s)
  - **Retries** (valeur par défaut = 3)
  - **Timeout time** (unité = 10ms, valeur par défaut = 1s)

### (5) Signalisation des défaillances et des états, diagnostics

- **Voir Chapitre 3**
- Signalisation des défaillances et de l'état de la passerelle par DEL sur le panneau avant
- **Voir Chapitre 5**
- Mode d'initialisation de la passerelle et description des informations de diagnostic

# 1. Introduction

## 1.2. Présentation de la passerelle LUF9

La passerelle LUF9 permet à un maître situé sur un réseau DeviceNet de dialoguer avec les esclaves d'un réseau Modbus RTU. Il s'agit d'un convertisseur de protocole générique qui fonctionne de manière transparente pour l'utilisateur.

Cette passerelle vous permet de relier de nombreux produits distribués par *Schneider Electric* à un réseau DeviceNet, tels que les départs-moteurs TeSys U, les variateurs Altivar et les démarreurs Altistart.

## 1.3. Terminologie

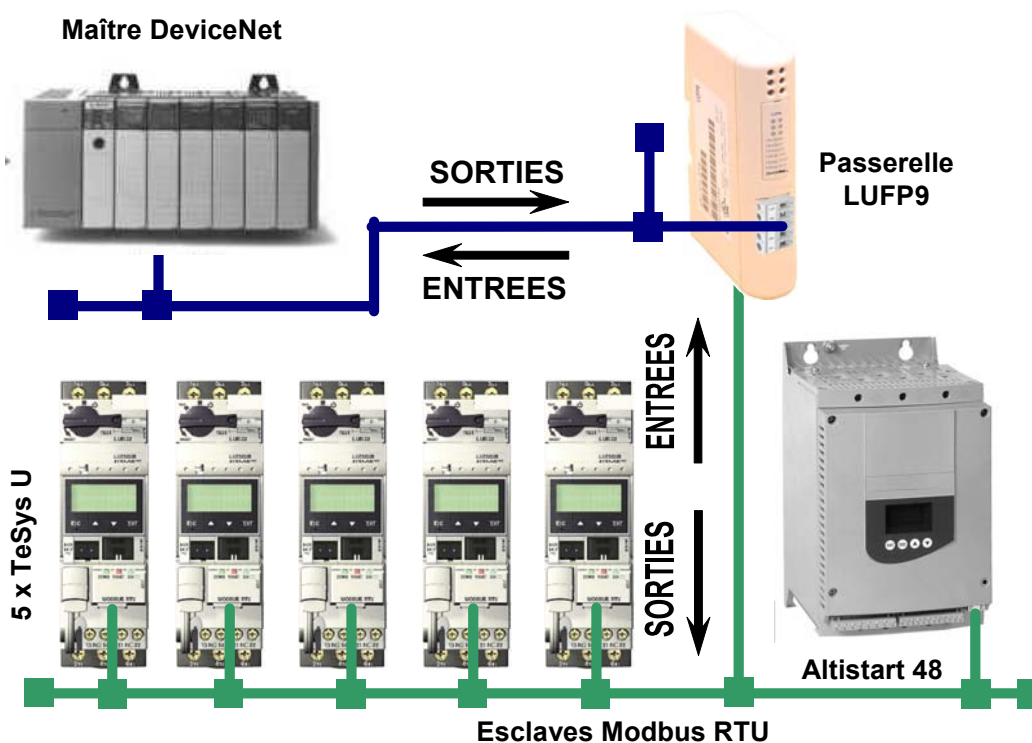
Tout au long de ce document, le terme « utilisateur » désigne la ou les personnes amenées à manipuler ou à se servir de la passerelle.

Le terme « RTU », qui caractérise le protocole de communication Modbus RTU, sera omis la plupart du temps. Par conséquent, le simple terme « Modbus » désignera le protocole de communication Modbus RTU.

Comme cela reste le cas pour tous les systèmes communicants, les termes « entrée » et « sortie » sont ambigus. Pour éviter toute confusion, nous utilisons une convention unique tout au long de ce document. Ainsi, les notions « entrée » et « sortie » sont toujours vues de l'automate, ou du maître / scanner DeviceNet.

Une « sortie » est donc un signal de commande envoyé à un esclave Modbus, tandis qu'une « entrée » est un signal de surveillance généré par ce même esclave Modbus.

Le schéma représenté ci-dessous symbolise le flux des « entrées » et des « sorties » échangées entre un maître DeviceNet et des esclaves Modbus RTU via la passerelle LUF9 :



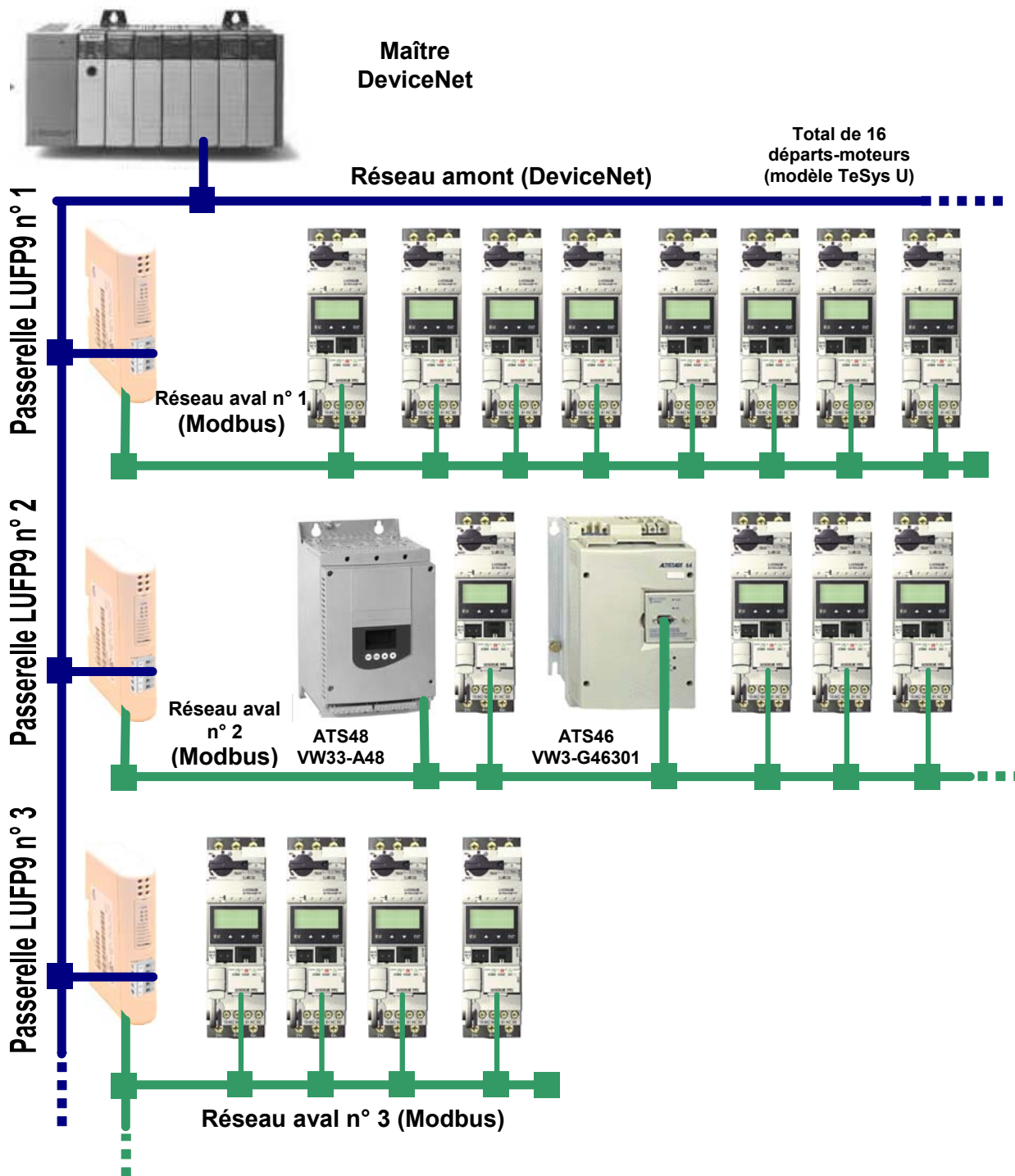
**REMARQUE** : Pour obtenir davantage d'informations concernant des termes spécifiques, reportez-vous au Glossaire disponible à la fin de ce guide.



# 1. Introduction

## 1.4. Présentation de l'architecture « système » des communications

Chaque passerelle DeviceNet / Modbus RTU LUF99 permet à un automate présent sur le réseau DeviceNet de commander, de contrôler et de configurer jusqu'à 8 esclaves Modbus. Il est possible de distribuer 50 commandes à 8 esclaves, sans contrainte de temps. Si le nombre d'esclaves Modbus est supérieur à 8, vous devrez avoir recours à un nombre approprié de passerelles LUF99.



# 1. Introduction

---

La passerelle LUF9 se comporte à la fois comme un esclave DeviceNet sur le réseau amont et comme un maître Modbus RTU sur le réseau aval.

Reportez-vous à l'Annexe A : Caractéristiques techniques, si vous désirez prendre connaissance des caractéristiques techniques de communication de la passerelle LUF9.

La passerelle peut effectuer ses échanges de données (entrées et sorties de tous types) avec les esclaves Modbus de manière cyclique, aperiodique ou événementielle. L'ensemble de ces échanges Modbus forment le "scanner Modbus" de la passerelle et on utilise le logiciel "ABC-LUF9 Config Tool" pour configurer les échanges de ce scanner. Chaque donnée échangée de cette manière est mise à la disposition du maître DeviceNet, qui pourra y accéder de diverses façons (échanges cycliques, aperiodiques ou événementiels).

**REMARQUE** : Si, par exemple, une communication est périodique sur le réseau Modbus, il n'est pas obligatoire que les données correspondantes soient échangées de manière périodique sur le réseau DeviceNet, et *vice versa*.

Le schéma situé sur la page précédente illustre la répartition de plusieurs esclaves sur trois réseaux avals Modbus RTU, chacun de ces réseaux étant interfacé avec l'automate maître DeviceNet à l'aide d'une passerelle LUF9.

## 1.5. Principe de configuration et de fonctionnement de la passerelle

La passerelle LUF9 fait partie d'une famille de produits (désignés par LUF9●) conçus pour répondre à des besoins génériques de connexion entre deux réseaux utilisant des protocoles de communication distincts.

Les éléments logiciels communs à toutes ces passerelles (outil de configuration, appelé « ABC-LUF9 Config Tool », et logiciel Modbus embarqué) cohabitent avec les spécificités du réseau amont de chacune d'elle (DeviceNet dans le cas de la passerelle LUF9) d'une manière générique. C'est l'une des raisons pour lesquelles l'interfaçage entre le réseau amont et le réseau Modbus est intégralement effectué via la mémoire physique de la passerelle.

⇒ Les échanges entre la passerelle (qui fait office de maître Modbus) et les esclaves Modbus sont entièrement configurés à l'aide de « ABC-LUF9 Config Tool ». Cet outil de configuration atteint un niveau de détail particulièrement élevé (temporisations des échanges, modes de communication, contenu des trames, etc.), ce qui rend son utilisation d'autant plus délicate. Un chapitre entier lui a donc été consacré dans le présent guide (chapitre 6).

# 1. Introduction

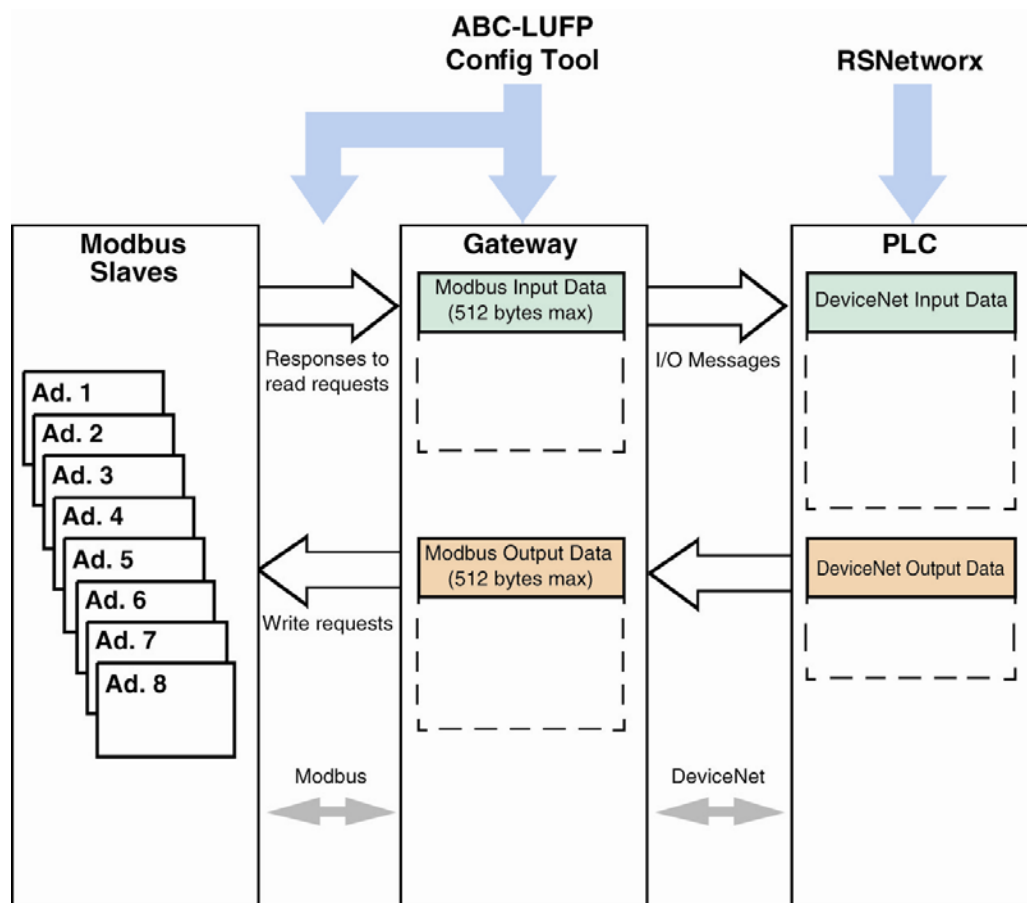
⇒ Chaque passerelle LUFF9 est livrée pré-configurée pour en simplifier l'utilisation et pour servir de base à une configuration qui répondrait au mieux aux attentes de l'utilisateur. Les opérations typiques applicables à cette configuration par défaut sont décrites dans le chapitre 6.

Le réseau DeviceNet est totalement dissocié du réseau Modbus. Les trames d'un réseau ne sont pas directement « traduites » par la passerelle pour générer des trames sur l'autre réseau. Au lieu de cela, les échanges entre le contenu de la mémoire de la passerelle et les esclaves Modbus forment un système indépendant de celui qui est chargé de la gestion des échanges entre cette même mémoire et le maître DeviceNet. Le système garantit la cohérence des données échangées dans la mémoire partagée.

Vous devez veiller à ce que la taille des données DeviceNet corresponde à la taille de la mémoire utilisée pour les échanges Modbus, car la passerelle configure ses échanges DeviceNet en se basant sur la mémoire utilisée par les trames Modbus. Si la taille ne correspond pas, la DEL Diag n°4 du bus de terrain clignote à une fréquence de 1 Hertz, les échanges Modbus cycliques sont activés et les registres Modbus accessibles en écriture sont définis sur 0.

L'exemple suivant illustre la gestion indépendante de chacun des deux réseaux :

## — Gestion des échanges Passerelle ↔ Esclaves Modbus —



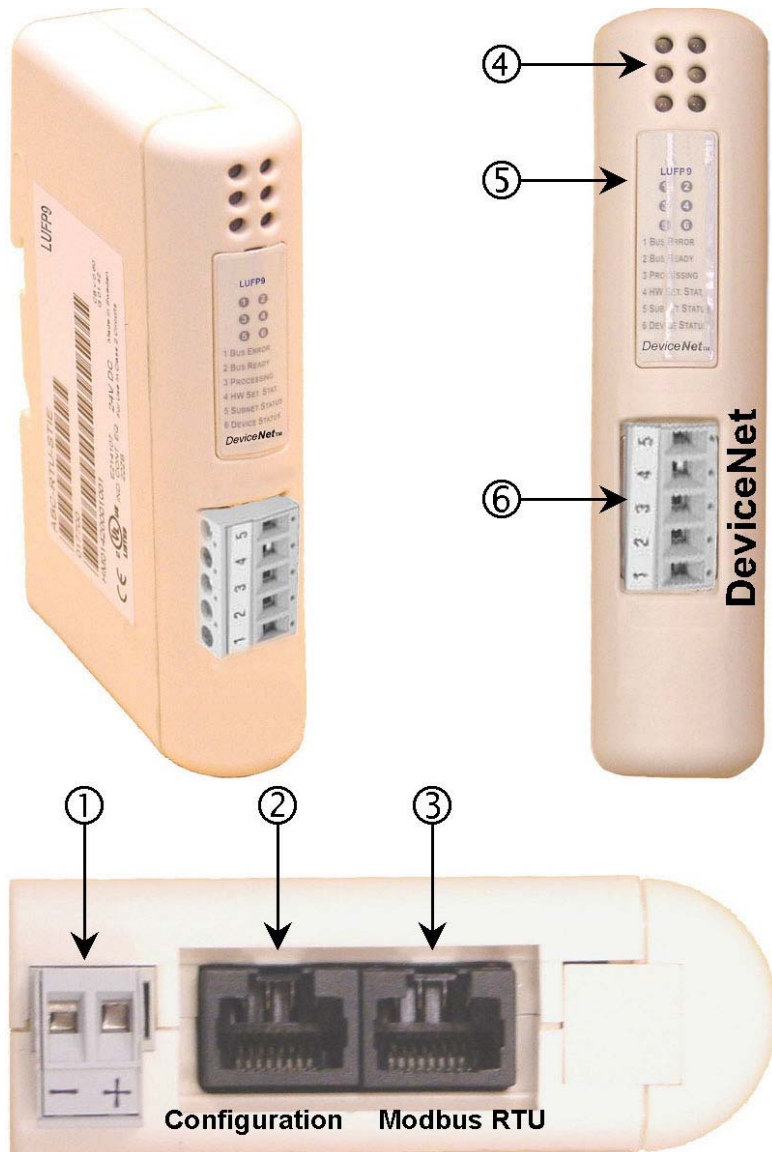
## 2. Mise en œuvre matérielle de la passerelle LUFPP9

### 2.1. Réception

Après ouverture de l'emballage, vérifiez la présence d'une passerelle LUFPP9 DeviceNet / Modbus RTU équipée d'un connecteur débrochable d'alimentation et d'un connecteur débrochable DeviceNet de type ouvert.

### 2.2. Présentation de la passerelle LUFPP9

Les câbles et autres accessoires de raccordement aux réseaux DeviceNet et Modbus doivent être commandés séparément.



#### Légende :

- ① Connecteur débrochable d'alimentation de la passerelle (== 24V).
- ② Connecteur RJ45 femelle pour liaison avec un PC doté du logiciel de configuration ABC-LUFPP Config Tool.
- ③ Connecteur RJ45 femelle du réseau aval Modbus RTU.
- ④ Six DEL de diagnostic.
- ⑤ Capot amovible dissimulant les commutateurs de configuration de la passerelle, représentés et décrits dans le chapitre 2.7. L'étiquette de description des DEL est collée sur ce même capot.
- ⑥ Connecteur DeviceNet femelle débrochable.

## 2. Mise en œuvre matérielle de la passerelle LUF9

La passerelle LUF9 permet des communications entre un réseau DeviceNet et des périphériques Modbus pour des applications industrielles d'automatisation et de contrôle. Comme pour tout composant utilisé dans un système de contrôle industriel, le concepteur doit évaluer les dangers potentiels découlant de l'utilisation de la passerelle LUF9 pour cette application.

### **⚠ AVERTISSEMENT**

#### **PERTE DE CONTRÔLE**

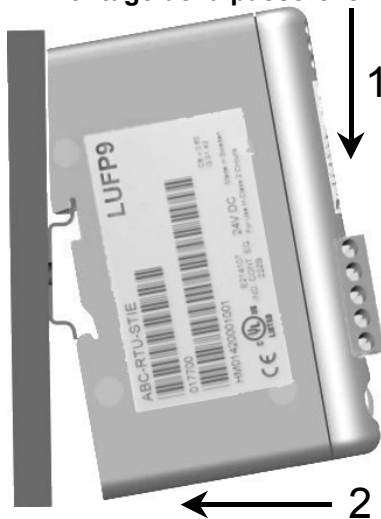
- Le concepteur de tout système de contrôle doit tenir compte des modes de défaillance potentielle des chemins de contrôle et, pour certaines fonctions de contrôle essentielles, prévoir un moyen d'atteindre un état sécurisé durant et après la défaillance d'un chemin. L'arrêt d'urgence et l'arrêt en cas de sur-course constituent des exemples de fonctions de contrôle essentielles.
- Des chemins de contrôle distincts ou redondants doivent être prévus pour les fonctions de contrôle essentielles.
- Les chemins de contrôle du système peuvent inclure des liaisons de communication. Il est nécessaire de tenir compte des conséquences des retards de transmission inattendus ou des défaillances d'une liaison.<sup>a</sup>
- Chaque mise en œuvre d'une passerelle LUF9 doit être testée de manière individuelle et approfondie afin de vérifier son fonctionnement avant de la mettre en service.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

<sup>a</sup> Pour plus d'informations, reportez-vous aux documents NEMA ICS 1.1 (nouvelle édition), « Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control » et NEMA ICS 7.1 (nouvelle édition), « Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems ».

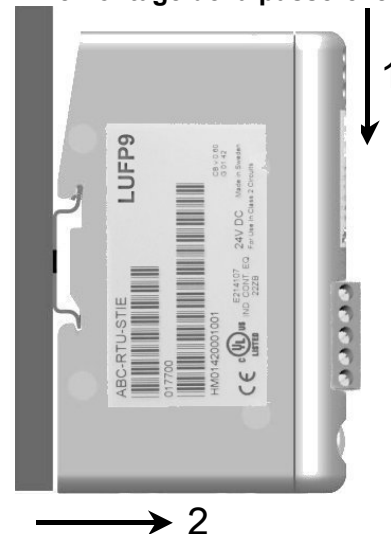
### 2.3. Montage de la passerelle sur rail DIN

**Montage de la passerelle**



Commencez par appliquer l'embase arrière de la passerelle sur la partie supérieure du rail, en poussant vers le bas (1) pour comprimer le ressort de la passerelle. Poussez ensuite la passerelle contre le rail DIN (2) jusqu'à ce que l'embase du boîtier de la passerelle s'emboîte sur le rail.

**Démontage de la passerelle**



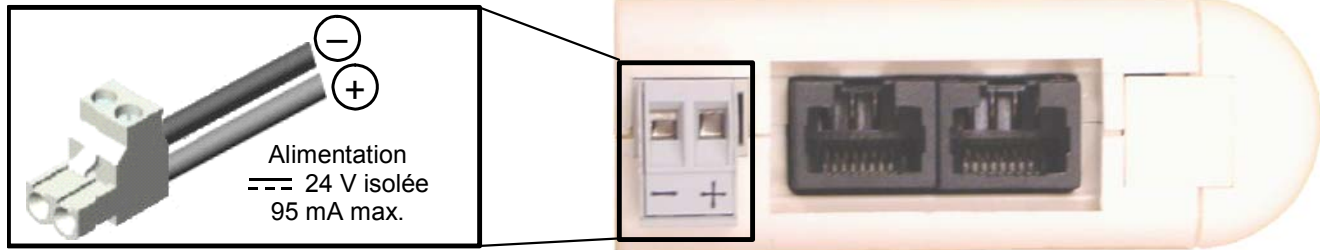
Commencez par pousser la passerelle vers le bas (1) pour comprimer le ressort de la passerelle. Tirez ensuite le bas du boîtier de la passerelle vers l'avant (2) jusqu'à ce que le dos du boîtier se déboîte du rail.

**REMARQUE :** Le ressort fait également office d'organe de mise à la terre de la passerelle (Protective Earth).

## 2. Mise en œuvre matérielle de la passerelle LUF9

### 2.4. Alimentation de la passerelle

Passerelle DeviceNet / Modbus RTU - Vue de dessous



### **⚠ AVERTISSEMENT**

#### **RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT**

N'utilisez pas l'alimentation 24 V CC fournie par le câble du réseau DeviceNet pour alimenter les passerelles LUF9, car la borne négative (—) de cette alimentation n'est pas nécessairement au même potentiel de mise à la terre que l'installation. L'utilisation d'une alimentation sans mise à la terre peut provoquer un fonctionnement imprévisible des périphériques LUF9.

Pour garantir un fonctionnement sûr, les passerelles LUF9 exigent une alimentation séparée, dont la borne négative (—) est connectée à la mise à la terre du système.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

Recommandations :

- Utilisez uniquement un câble en cuivre (CU) 60/75 ou 75xC.
- Le couple de serrage des bornes doit être compris entre 0,5 et 0,8 Nm.

### 2.5. Raccordement de la passerelle au réseau Modbus

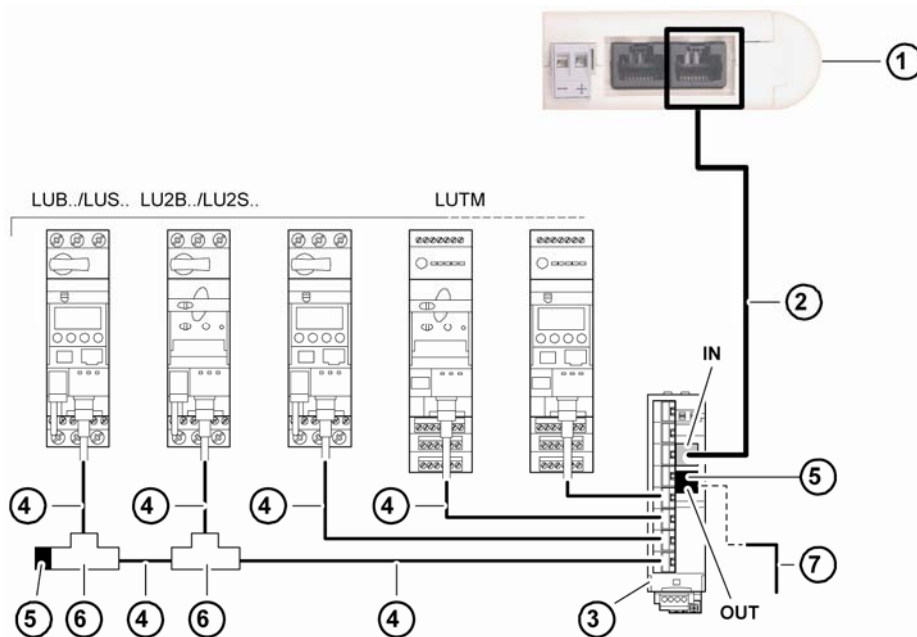
Trois exemples types de raccordement Modbus de la passerelle et de ses esclaves sont présentés ci-après. Il existe de nombreuses autres possibilités de raccordement Modbus, mais elles ne font pas l'objet de ce document.

## 2. Mise en œuvre matérielle de la passerelle LUF9

### 2.5.1. Exemples de topologies de raccordement Modbus

- Topologie « bus » avec répartiteur LU9 GC3.

Les branchements sont schématisés ci-dessous :



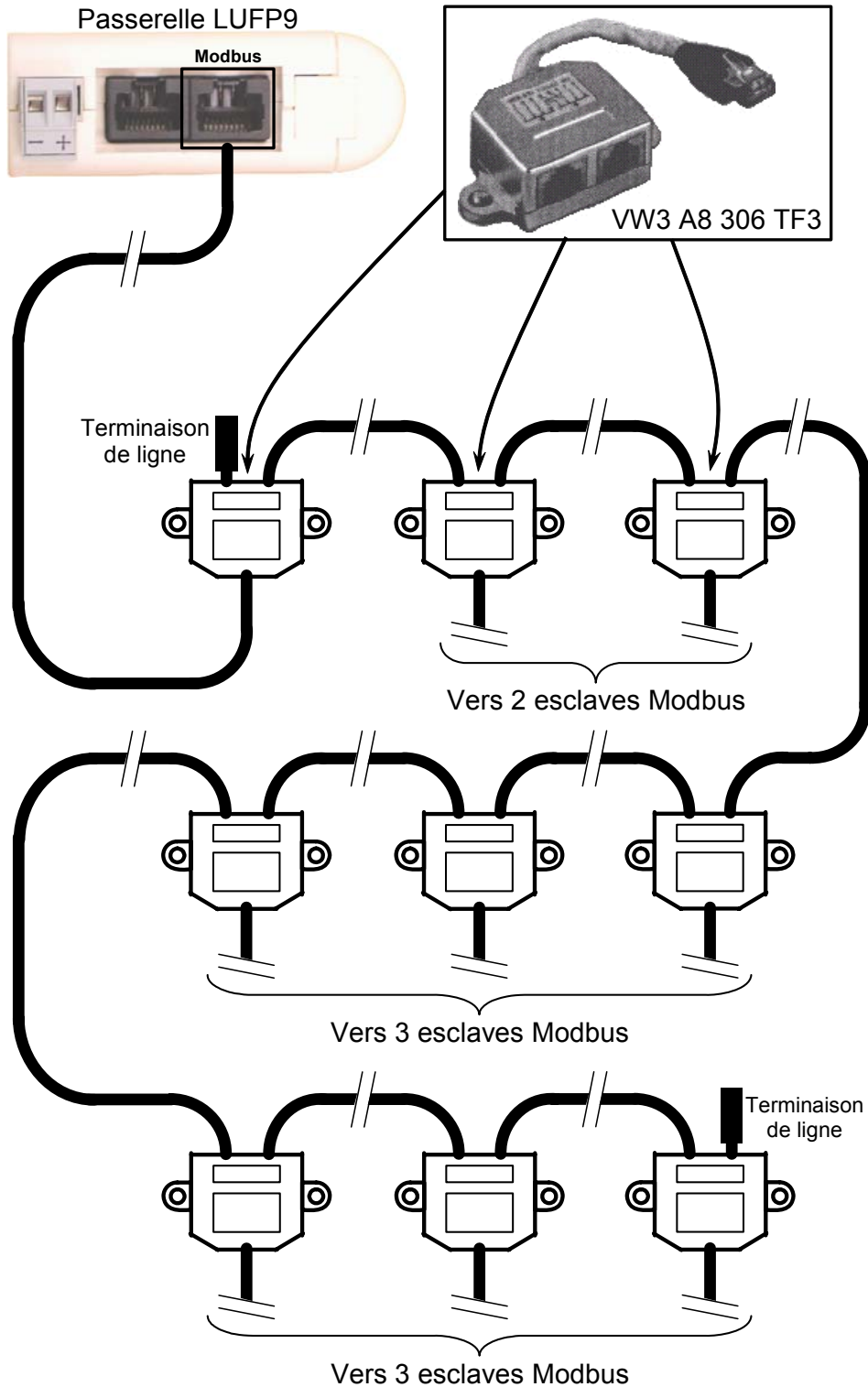
- 1 Passerelle LUF9
- 2 Câble Modbus
- 3 Répartiteur Modbus LU9 GC3
- 4 Câbles Modbus VW3 A8 306 R\*\*
- 5 Terminaisons VW3 A8 306 R
- 6 Boîtes de dérivation T Modbus VW3A8306TF\*\* (avec câble)
- 7 Câble Modbus (vers un autre répartiteur) TSX CSA\*00 (en remplacement du (5))

**REMARQUE :** Il est recommandé d'installer une terminaison de part et d'autre du bus afin d'éviter tout dysfonctionnement sur le bus de communication. Cela signifie qu'aucun connecteur ne doit être libre sur un té et que ce dernier est raccordé à un esclave ou au maître, ou qu'une terminaison est installée.

**REMARQUE :** Il est important de raccorder le bus à l'entrée (IN) du répartiteur. La sortie (OUT) est utilisée pour le raccordement à un autre répartiteur.

## 2. Mise en œuvre matérielle de la passerelle LUF9

**Topologie « bus » avec boîtes de dérivation T VW3 A8 306 TF3 :** Cette topologie utilise des boîtes de dérivation T VW3 A8 306 TF3 afin de relier chacun des esclaves Modbus au tronçon principal du réseau Modbus. Chaque boîte doit être placée à proximité immédiate de l'esclave Modbus auquel elle est associée. Le câble du tronçon principal du réseau Modbus doit être doté de connecteurs RJ45 mâles (tel que le câble VW3 A8 306 R•• utilisé pour la Topologie « étoile »). Le cordon reliant la boîte de dérivation à l'esclave ou à la passerelle Modbus fait partie intégrante de cette même boîte. Les branchements sont schématisés ci-dessous :

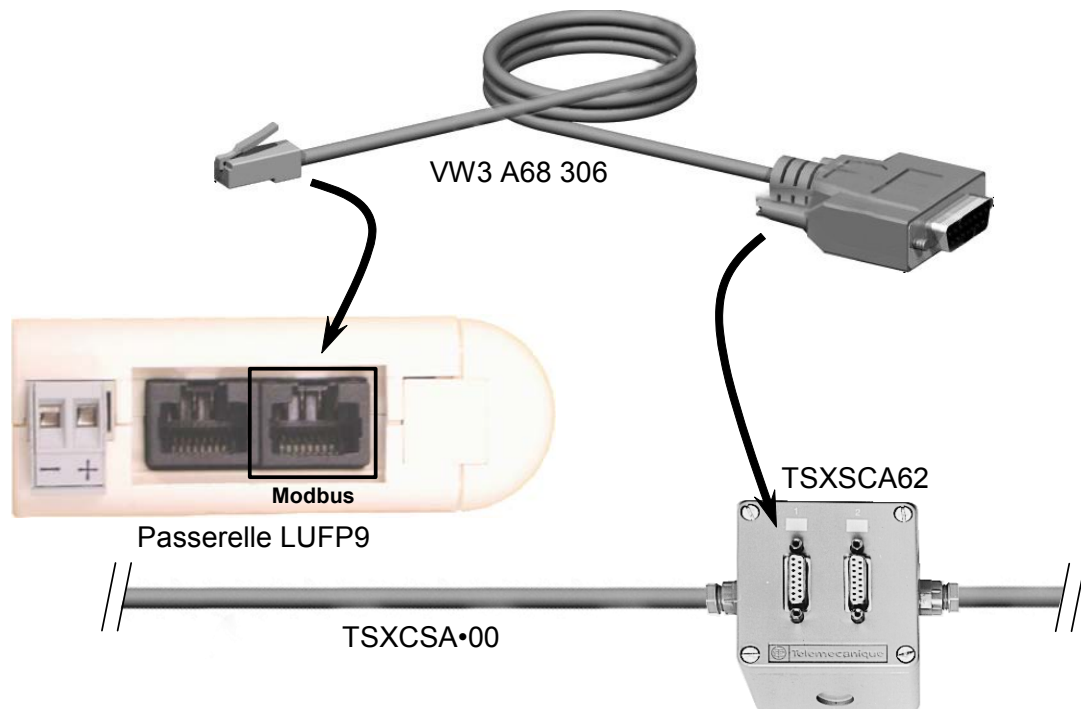




## 2. Mise en œuvre matérielle de la passerelle LUF9

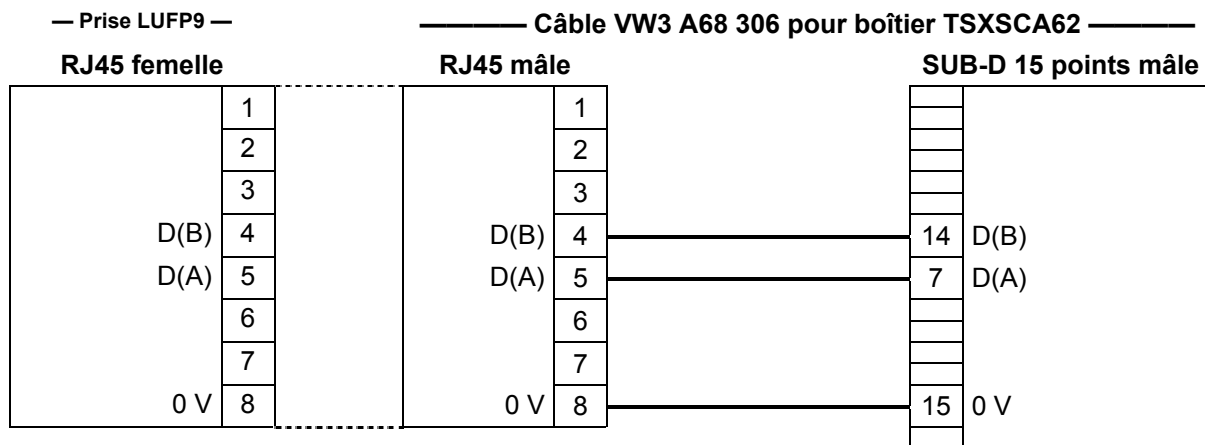
- **Topologie « bus » avec boîtes de dérivation SCA** : Cette topologie est similaire à la précédente, sauf qu'elle utilise les connecteurs de l'abonné TSXSCA62 et/ou les connecteurs de l'abonné TSXCA50. Il est recommandé d'utiliser un câble de connexion VW3 A68 306 et des câbles Modbus TSXCSA•00. Raccordez le connecteur RJ45 du câble VW3 A68 306 au connecteur Modbus de la passerelle LUF9.

Les branchements sont schématisés ci-dessous :



### 2.5.2. Brochage

En plus du brochage de la prise située sur la passerelle, celui du câble VW3 A68 306 est également présenté ci-dessous, car il est le seul câble Modbus à ne pas utiliser exclusivement une connectique en RJ45.



## 2. Mise en œuvre matérielle de la passerelle LUF9

### 2.5.3. Recommandations de câblage du réseau Modbus

- Utilisez un câble blindé avec 2 paires de conducteurs torsadés,
- reliez les potentiels de référence entre eux,
- longueur maximale de la ligne : 1 000 mètres
- longueur maximale d'une dérivation : 20 mètres
- ne connectez pas plus de 9 stations sur un bus (esclaves et passerelle LUF9 confondus),

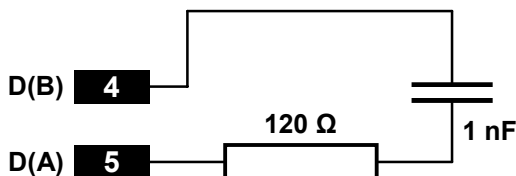
### **⚠ AVERTISSEMENT**

#### **RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT**

Ne connectez pas plus de 9 stations au bus de terrain Modbus (8 esclaves et une passerelle). Même si la passerelle semble fonctionner correctement avec plus de 9 périphériques, il est probable qu'un ou plusieurs périphériques communiquent par intermittence uniquement, provoquant un comportement imprévisible du système.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

- cheminement du câble : éloignez le bus des câbles d'alimentation (30 cm au minimum), effectuez les croisements à angle droit si nécessaire et raccordez le blindage du câble à la masse de chaque équipement,
- adaptez la ligne à ses deux extrémités à l'aide d'une terminaison de ligne de type RC (voir schéma et terminaison VW3 A8 306 RC ci-dessous).



— Adaptation de fin de ligne recommandée aux 2 extrémités — — Terminaison de ligne VW3 A8 306 RC —

### **⚠ AVERTISSEMENT**

#### **TERMINAISON DE LIGNE MODBUS A L'AIDE DE LA METHODE PAR RESISTANCE UNIQUEMENT**

Utilisez uniquement des terminaisons de câble Modbus RC (Resistance-Capacitance) avec la passerelle LUF9. Les passerelles LUF9 sont conçues pour prendre en charge des équipements clients qui ne fonctionneront pas correctement sans utiliser de terminaisons de câble Modbus de type RC.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

Pour faciliter le raccordement des équipements selon les topologies décrites dans le chapitre 2.5.1, divers accessoires sont proposés au catalogue *Schneider Electric* :

## 2. Mise en œuvre matérielle de la passerelle LUP9

---

### 1) Répartiteurs, dérivation et terminaisons de ligne :

Cette boîte passive comporte 8 connecteurs femelles RJ45. Chacun de ces connecteurs peut être connecté à un esclave Modbus, à un maître Modbus, à un autre répartiteur Modbus ou à une terminaison de ligne.

Répartiteur LU9GC03 .....  
(topologie « bus »)

Cette boîte passive comporte un cordon court avec connecteur RJ45 mâle permettant de le brancher directement sur un esclave Modbus, sans devoir utiliser un câble distinct. Elle est équipée de 2 connecteurs femelles RJ45 pour le raccordement de deux câbles Modbus de type VW3 A8 306 R••.

Boîte de dérivation T VW3 A8 306 TF3 ..  
(topologie « bus » avec boîtes de dérivation T VW3 A8 306 TF3)

Cette boîte passive comporte un circuit imprimé équipé de borniers à vis et permet le raccordement de 2 abonnés sur le bus (2 connecteurs SUB-D 15 points femelles). Elle inclut la terminaison lorsque le connecteur se situe en bout de ligne. Elle est équipée de 2 borniers à vis pour le raccordement de deux câbles Modbus double paire torsadée.

Prise abonnés 2 voies TSXSCA62 .....  
(topologie « bus » avec boîtes de dérivation)

Cette boîte passive permet de connecter une unité Modbus à un bornier à vis. Elle inclut la terminaison lorsque le connecteur se situe en bout de ligne. Elle est équipée de 2 borniers à vis pour le raccordement de deux câbles Modbus double paire torsadée.

Boîte de dérivation SCA TSXCA50.....  
(topologie « bus » avec boîtes de dérivation SCA)

Chacune de ces deux boîtes passives de couleur rouge est un connecteur RJ45 mâle de 3 cm de long contenant une terminaison de ligne RC (voir schéma et illustration ci-dessus). Seule l'abréviation « RC » est portée sur ces boîtiers.

Double terminaison VW3 A8 306 RC.....  
(toutes topologies)

---

### 2) Câbles :

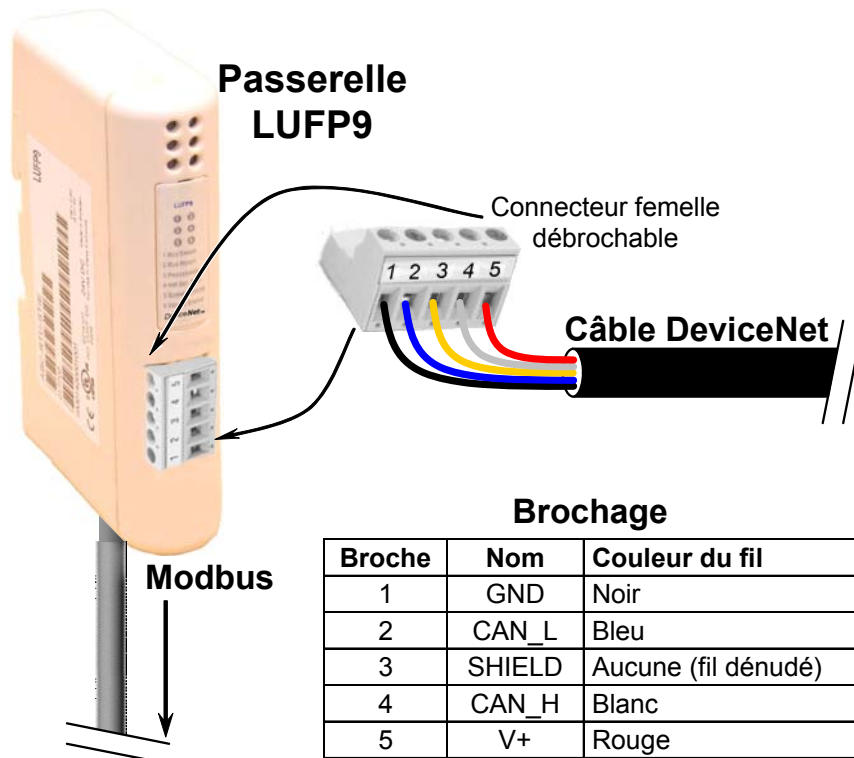
- Câble Modbus VW3 A8 306 R••.....Câble blindé doté d'un connecteur mâle RJ45 à chacune (topologie « bus » avec boîtes de dérivation SCA) de ses extrémités.
- Câble Modbus VW3 A8 306.....Câble blindé doté d'un connecteur mâle RJ45 et d'un (topologie « bus » avec boîtes de dérivation SCA) connecteur SUB-D 15 points mâle. Il sert à raccorder un abonné Modbus (esclave ou maître) à une boîte TSXSCA62 ou TSXCA50.
- Câble Modbus double paire torsadée blindée.....Câble nu (sans connecteurs) destiné à constituer le (topologie « bus » avec boîtes de dérivation) tronçon principal du réseau Modbus. Trois références sont disponibles : TSXCSA100 (100 m), TSXCSA200 (200 m) et TSXCSA500 (500 m).

## 2. Mise en œuvre matérielle de la passerelle LUF9

### 2.6. Connexion de la passerelle LUF9 au réseau DeviceNet

Si la passerelle LUF9 est physiquement située à l'une des deux extrémités du réseau DeviceNet, il est nécessaire de brancher une terminaison de ligne aux bornes de son connecteur DeviceNet.

La résistance de cette terminaison de ligne doit être égale à  $121 \Omega$  et elle doit être connectée entre les broches 2 et 4 du connecteur de la passerelle, c'est-à-dire entre les signaux CAN\_L et CAN\_H.



## 2. Mise en œuvre matérielle de la passerelle LUPF9

### 2.7. Configuration des fonctions de communication DeviceNet

Cette configuration doit être effectuée lorsque la passerelle est hors tension.

#### **⚠ ATTENTION**

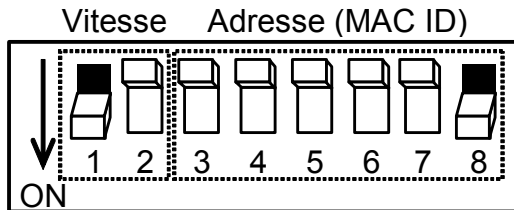
##### **OUVERTURE DU CAPOT DE LA PASSERELLE LUPF• SOUS TENSION**

L'alimentation de la passerelle doit être coupée avant d'ouvrir le capot. Une fois le capot retiré, veillez à ne toucher ni les circuits électriques, ni les composants électroniques, car vous risqueriez d'endommager l'appareil.

**Le non-respect de ces instructions peut entraîner des blessures ou des dommages matériels.**

Le bloc de commutateurs permettant de configurer les fonctions de communication DeviceNet est dissimulé derrière le capot ⑤ de la passerelle (voir illustration du chapitre 2.2). Pour retirer ce capot, il suffit de glisser la pointe d'un petit tournevis entre le sommet du capot et le boîtier de la passerelle, puis de le retirer avec précaution.

Le bloc de commutateurs est schématisé ci-dessous, chaque commutateur étant symbolisé dans sa position de réglage usine :



Un commutateur est à l'état 0 lorsqu'il est en position OFF et à l'état 1 lorsqu'il est en position ON.

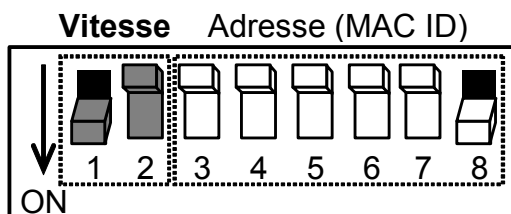
REMARQUE : Toute modification des fonctions de communication de la passerelle ne sera prise en compte qu'à la prochaine mise sous tension de la passerelle.

#### 2.7.1. Codage de la vitesse DeviceNet

La vitesse de communication de la passerelle sur le réseau DeviceNet doit être identique à celle du maître DeviceNet. Dans le cas contraire, une erreur de configuration surviendra.

Le réglage usine est 500 kbits/s.

La valeur de cette vitesse dépend du positionnement des commutateurs 1 et 2.

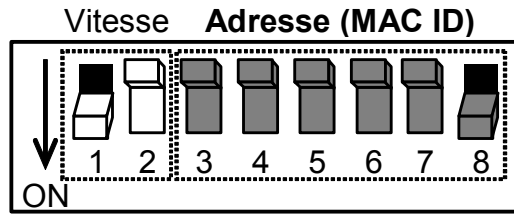


Commutateurs 1 2 3 4 5 6 7 8	Débit DeviceNet
0 0 x x x x x x	125 kbits/s
0 1 x x x x x x	250 kbits/s
1 0 x x x x x x	500 kbits/s
1 1 x x x x x x	Configuration invalide

## 2. Mise en œuvre matérielle de la passerelle LUF9

### 2.7.2. Codage de l'adresse de la passerelle

La passerelle LUF9 est identifiée sur le bus DeviceNet par son adresse (ou « Mac ID »), comprise entre 0 et 63.



L'adresse DeviceNet de la passerelle dépend du positionnement des commutateurs 3 à 8. Elle correspond au nombre binaire donné par la position ON (1) ou OFF (0) de ces 6 commutateurs.

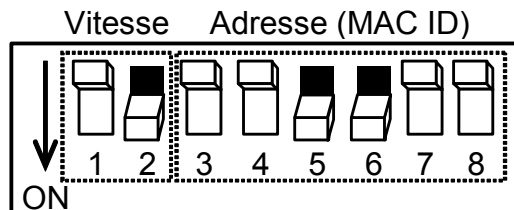
Commutateurs	Adresse
xx000000	0
xx000001	1
xx000010	2
xx000011	3
xx000100	4
xx000101	5
xx000110	6
xx000111	7
xx001000	8
xx001001	9
xx001010	10
xx001011	11
xx001100	12
xx001101	13
xx001110	14
xx001111	15
xx010000	16
xx010001	17
xx010010	18
xx010011	19
xx010100	20
xx010101	21

Commutateurs	Adresse
xx010110	22
xx010111	23
xx011000	24
xx011001	25
xx011010	26
xx011011	27
xx011100	28
xx011101	29
xx011110	30
xx011111	31
xx100000	32
xx100001	33
xx100010	34
xx100011	35
xx100100	36
xx100101	37
xx100110	38
xx100111	39
xx101000	40
xx101001	41
xx101010	42
xx101011	43

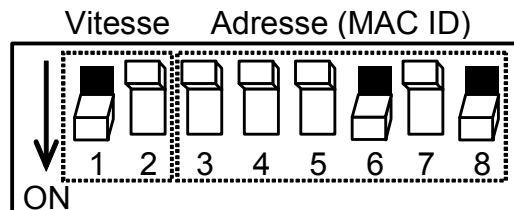
Commutateurs	Adresse
xx101100	44
xx101101	45
xx101110	46
xx101111	47
xx110000	48
xx110001	49
xx110010	50
xx110011	51
xx110100	52
xx110101	53
xx110110	54
xx110111	55
xx111000	56
xx111001	57
xx111010	58
xx111011	59
xx111100	60
xx111101	61
xx111110	62
xx111111	63

### 2.7.3. Exemples de configuration de la passerelle

Vitesse = 250 kbits/s  
Adresse = 12

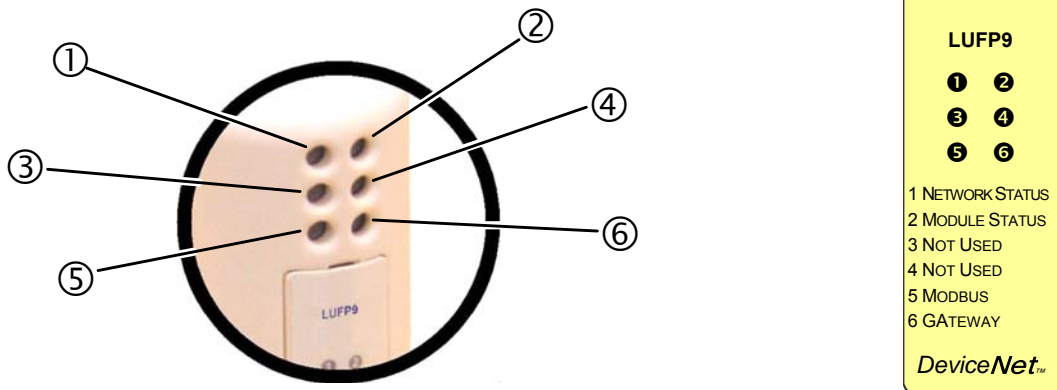


Vitesse = 500 kbits/s  
Adresse = 5



### 3. Signalisation

Les 6 DEL de la passerelle et l'étiquette descriptive figurant sur le capot amovible permettent de diagnostiquer l'état de la passerelle :



DEL	DEL → Etat de la passerelle
<b>1</b> NETWORK STATUS	Eteinte : Passerelle non connectée au bus DeviceNet.
	<b>Verte : Passerelle connectée au bus DeviceNet : Connexion établie</b>
	Rouge : Echec fatal lors de la connexion au bus DeviceNet
	Clignotante (vert) : Passerelle connectée au bus DeviceNet : Connexion non établie
<b>3</b> NOT USED	Clignotante (rouge) : Timeout de connexion au bus DeviceNet ↳ <i>La durée de ce timeout est définie par le maître DeviceNet</i>
	<b>Eteinte : —</b>
<b>5</b> MODBUS	Eteinte : Pas d'alimentation
	Clignotante (vert) : Pas de communications Modbus
	<b>Verte : Communications Modbus OK</b>
	Rouge : - Perte de communication avec un ou plusieurs esclaves Modbus (pas de réponse de l'esclave) (1) - Code d'exception provenant d'une commande ou d'une transaction

DEL	DEL → Etat de la passerelle
<b>2</b> MODULE STATUS	Eteinte : Pas d'alimentation
	Rouge : Défaut irrémédiable
	<b>Verte : Passerelle opérationnelle</b>
	Clignotante (rouge) : Défaut
<b>4</b> NOT USED	<b>Eteinte : —</b>
<b>6</b> GATEWAY	Eteinte : Pas d'alimentation
	Clignotante (rouge/vert) : Configuration absente / non valide ↳ <i>Utilisez ABC-LUFF Config Tool pour charger une configuration correcte.</i>
	Verte : Passerelle en cours d'initialisation et de configuration
	<b>Clignotante (vert) : Passerelle en cours de fonctionnement : Configuration OK</b>

### 3. Signalisation

---

- (1) La DEL MODBUS ⑤ devient rouge lorsqu'un ou plusieurs esclaves Modbus ne répondent pas à la passerelle de façon attendue. Ce comportement peut être dû à :
- une perte de communication (un câble est endommagé ou déconnecté, par exemple),
  - des valeurs d'écriture incorrectes dans les sorties qui correspondent aux deux services apériodiques de lecture/écriture (voir chapitre 4.3).

**REMARQUE** : Lorsque la DEL MODBUS ⑤ clignote en rouge en raison d'une simple perte de communication, elle redeviendra verte lorsque les communications sont restaurées. Lorsque la DEL (5) clignote en rouge en raison de l'utilisation de valeurs incorrectes avec les services apériodiques de lecture/écriture, la seule façon d'effacer cette erreur est de réutiliser ces services apériodiques avec des valeurs correctes.

**REMARQUE** : Si la DEL GATEWAY ⑥ clignote suivant une séquence commençant par un ou plusieurs flashes rouges, il est conseillé de noter l'ordre du déroulement de cette séquence et de communiquer ces renseignements au service de support de Schneider Electric. Dans certains cas, le problème se résout simplement par la mise hors tension de la passerelle puis sa remise sous tension.



# 4. Mise en œuvre logicielle de la passerelle

## 4.1. Introduction

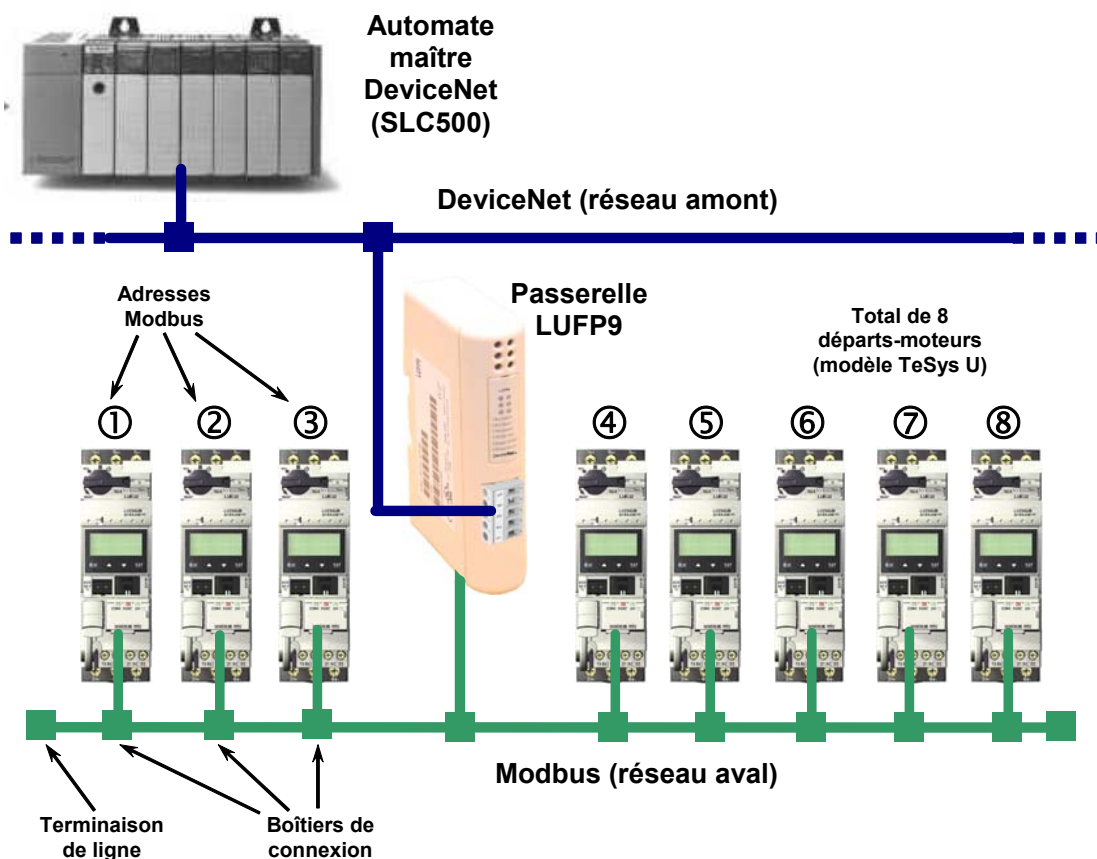
Ce chapitre présente une mise en œuvre rapide de la passerelle LUF9, grâce à l'utilisation de sa configuration par défaut, l'ensemble des passerelles LUF9 étant livrées pré-configurées.

**REMARQUE** : La configuration a été définie pour 8 départs-moteurs. Si vous en utilisez moins de 8, reportez-vous au chapitre 6.

La configuration par défaut proposée par Schneider Electric a pour objectif de fournir un bon point de départ pour les clients utilisant des départs-moteurs TeSys U, ainsi que de limiter les modifications de la configuration nécessaires pour la plupart des installations. La configuration par défaut permet la mise en œuvre de la passerelle à l'aide d'un outil de configuration pour automate maître DeviceNet. Cependant, il incombe à l'utilisateur uniquement de s'assurer que la configuration par défaut, ou toute autre configuration, est sûre et appropriée pour ses installations et l'usage prévu.

### 4.1.1. Architecture système

La configuration par défaut d'une passerelle LUF9 lui permet d'effectuer la commande, la surveillance et le paramétrage de 8 départs-moteurs TeSys U :



Reportez-vous au chapitre 2, pour la mise en œuvre matérielle de la configuration par défaut.

## 4. Mise en œuvre logicielle de la passerelle

---

### 4.1.2. Configuration des départs-moteurs

Chaque départ-moteur doit être configuré de la manière suivante :

Protocole :	Modbus RTU esclave	Bits de start	1
Adresse Modbus	1 à 8	Parité	Aucune
Vitesse de transmission	19 200 bits/s	Bit de parité	0
Bits de données	8	Bits de stop	1

Dans le cas d'un départ-moteur TeSys U doté d'un module de communication Modbus (LULC03\*), les paramètres de configuration de la liaison RS485 sont automatiquement détectés ; seule doit être configurée l'adresse Modbus du départ-moteur.

### 4.1.3. Temps de cycle Modbus

La configuration par défaut de la passerelle LUF9 impose un temps de cycle de 300 ms aux commandes Modbus. Ce temps de cycle correspond au temps d'interrogation nécessaire pour couvrir les 8 départs-moteurs.

### 4.1.4. Gestion des modes dégradés avec la configuration par défaut de la passerelle

La gestion des modes dégradés avec la configuration par défaut de la passerelle est décrite ci-dessous, mais elle ne tient pas compte de l'automate utilisé, ni du scanner DeviceNet. Reportez-vous au chapitre 6.12.2.1, si vous souhaitez gérer les modes dégradés de toute autre configuration.

#### 4.1.4.1. Description des options de mode dégradé de la passerelle

##### Offline options for fieldbus

Cette option affecte les données envoyées à un esclave Modbus si aucune communication ne provient du maître DeviceNet.

Elle est définie au niveau de la requête de chaque commande ou transaction envoyée aux différents esclaves.

Cette option peut prendre 3 valeurs :

Clear : Toutes les données envoyées à l'esclave Modbus concerné ont la valeur 0.

Freeze : Toutes les données envoyées conservent leur valeur actuelle.

No scanning : La requête n'est plus transmise.

Avec la configuration par défaut de la passerelle :

L'option « Clear » est sélectionnée pour les échanges périodiques.

L'option « No scanning » est sélectionnée pour les échanges apériodiques.

Cela signifie que les registres Tesys Command et Status continuent à être actualisés, mais la mémoire de sortie associée (registres de commande du Tesys U) est forcée à 0, et la mémoire d'entrée (registres d'état du Tesys U) fonctionne normalement, alors que les échanges Modbus apériodiques sont interrompus.

##### Timeout time

Cette option définit le délai pendant lequel la passerelle attend une réponse avant d'essayer de renvoyer la même requête ou de déconnecter l'esclave et de le déclarer manquant.

Elle est définie au niveau de la requête de chaque commande ou transaction envoyée aux différents esclaves.

Avec la configuration par défaut de la passerelle, ce délai est de 300 ms.

## 4. Mise en œuvre logicielle de la passerelle

---

### Retries

Cette option détermine le nombre de retransmissions effectuées par la passerelle en cas d'absence de réponse de l'esclave.

Elle est définie au niveau de la requête de chaque commande ou transaction envoyée aux différents esclaves. Avec la configuration par défaut de la passerelle, cette option est définie sur la valeur 3.

### Reconnect time

Cette option définit le temps que la passerelle attendra avant d'essayer de communiquer à nouveau avec un esclave Modbus précédemment déclaré manquant.

Elle est définie au niveau de la requête de chaque commande ou transaction envoyée aux différents esclaves. Avec la configuration par défaut de la passerelle, ce délai est de 10 secondes.

## AVERTISSEMENT

### RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

Durant le délai de reconnexion, il est impossible de contrôler un esclave (lecture/écriture) via le bus. Selon les caractéristiques de l'esclave et la configuration du chien de garde, l'esclave peut conserver le même état ou prendre une position de repli.

Afin d'éviter tout fonctionnement imprévu de l'appareil, vous devez connaître l'état possible d'un esclave et adapter le délai de timeout et de reconnexion en fonction de la vitesse d'envoi de la requête.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

### Offline options for sub-network

Cette option affecte les données envoyées au scanner DeviceNet lorsque aucune réponse ne provient d'un esclave.

Elle est définie au niveau de la réponse de chaque commande ou transaction envoyée depuis les différents esclaves.

Cette option peut prendre 2 valeurs :

Clear : Toutes les données envoyées au scanner DeviceNet ont la valeur 0.

Freeze : Toutes les données envoyées au scanner DeviceNet conservent leur valeur actuelle.

Avec la configuration par défaut de la passerelle, l'option « Clear » est sélectionnée et les registres d'état du Tesys U ainsi que les données d'entrée apériodiques sont forcées à 0.

#### 4.1.4.2. Description du mode dégradé

Cette description prend en compte les éléments suivants :

Le processeur de l'automate

Le scanner DeviceNet

La passerelle LUFP9

Les démarreurs-contrôleurs Tesys U.

## 4. Mise en œuvre logicielle de la passerelle

---

### Arrêt ou défaillance du processeur de l'automate

<b>Réponse du processeur de l'automate</b>
Sorties : Erreur logicielle, réinitialisation des sorties sur leur état par défaut ou conservation de leur état actuel, selon la configuration. Erreur matérielle (EEPROM ou défaillance matérielle), état de sortie indéterminé. Entrées : L'automate cesse de répondre aux entrées quel que soit l'état d'erreur.
<b>Réponse du scanner DeviceNet</b>
En fonction de la configuration du scanner : le scanner cesse de communiquer avec la passerelle LUF9, force les sorties DeviceNet sur la valeur 0 et actualise les entrées ou maintient les sorties DeviceNet sur leur dernière position et actualise les entrées.
<b>Réponse de la passerelle LUF9</b>
Si le scanner cesse de communiquer avec la passerelle : les échanges Modbus périodiques continuent de s'exécuter avec la mémoire de sortie associée forcée sur la valeur 0, la mémoire d'entrée continue à être actualisée, les échanges Modbus aperiodiques sont interrompus.
Si le scanner force les sorties DeviceNet sur la valeur 0 et actualise les entrées : les échanges Modbus périodiques continuent de s'exécuter avec les sorties définies sur 0, la mémoire d'entrée continue à être actualisée, les échanges Modbus aperiodiques sont interrompus.
Si le scanner maintient les sorties DeviceNet et actualise les entrées : les échanges Modbus périodiques continuent de s'exécuter avec la mémoire de sortie associée maintenue sur sa dernière position, la mémoire d'entrée continue à être actualisée, les échanges Modbus aperiodiques sont interrompus.
<b>Réponse du Tesys U</b>
Si le scanner cesse de communiquer ou force les sorties sur 0 : les échanges Modbus périodiques continuent de s'exécuter les registres de commande sont définis sur 0 et les moteurs sont arrêtés, le registre d'état est transmis à la passerelle, les échanges Modbus aperiodiques sont interrompus.
Si le scanner maintient les mots de sortie DeviceNet et actualise les mots d'entrées : les échanges Modbus périodiques continuent de s'exécuter, les registres de commande conservent leur dernière valeur et les moteurs restent dans le même état, les données du registre d'état sont transmises à la passerelle, les échanges Modbus aperiodiques sont interrompus.

## 4. Mise en œuvre logicielle de la passerelle

---

### Arrêt ou défaillance du scanner DeviceNet

<b>Réponse du processeur de l'automate</b>
Le processeur de l'automate fournit à l'application plusieurs erreurs et/ou objets de diagnostic au cas où le scanner DeviceNet cesserait de fonctionner ou connaîtrait une défaillance (entrée/sortie non valide). Reportez-vous au manuel d'utilisation de l'automate pour consulter leur description. Ces informations doivent être gérées dans l'application de l'automate.
<b>Réponse du scanner DeviceNet</b>
Si le scanner DeviceNet est arrêté (commande provenant de l'application) : le scanner cesse de communiquer avec la passerelle LUF9.
Si le scanner DeviceNet connaît une défaillance, le scanner cesse de communiquer avec le processeur et la passerelle LUF9.
<b>Réponse de la passerelle LUF9</b>
Avec la configuration par défaut de la passerelle (Offline option for fieldbus) : les échanges Modbus périodiques continuent de s'exécuter, avec la mémoire de sortie associée forcée sur la valeur 0, la mémoire d'entrée continue à être actualisée, les échanges Modbus apériodiques sont interrompus.
<b>Réponse du Tesys U</b>
les échanges Modbus périodiques continuent de s'exécuter : les registres de commande sont définis sur 0 et les moteurs sont arrêtés, les données du registre d'état sont transmises à la passerelle, les échanges Modbus apériodiques sont interrompus.

### Passerelles LUF9 déconnectées du côté DeviceNet

<b>Réponse de l'automate</b>
Le processeur de l'automate fournit plusieurs objets d'erreur et de diagnostic provenant du scanner DeviceNet en cas de déconnexion d'un esclave de l'application : Reportez-vous au manuel d'utilisation de l'automate pour consulter leur description. Ces informations doivent être gérées dans l'application de l'automate.
<b>Réponse du scanner DeviceNet</b>
Le scanner DeviceNet fournit au processeur différents objets d'erreur et de diagnostic en cas de déconnexion d'un esclave DeviceNet.
<b>Réponse de la passerelle LUF9</b>
Avec la configuration par défaut de la passerelle (Offline option for fieldbus) : les échanges Modbus périodiques continuent de s'exécuter, avec la mémoire de sortie associée forcée sur la valeur 0, la mémoire d'entrée continue à être actualisée, les échanges Modbus apériodiques sont interrompus.
<b>Réponse du Tesys U</b>
Les échanges Modbus périodiques continuent de s'exécuter : les registres de commande sont définis sur 0 et les moteurs sont arrêtés, les données du registre d'état sont transmises à la passerelle, les échanges Modbus apériodiques sont interrompus.

## 4. Mise en œuvre logicielle de la passerelle

---

### Défaillance des passerelles LUF9

<b>Réponse de l'automate</b>
Le processeur de l'automate fournit plusieurs objets d'erreur et de diagnostic provenant du scanner DeviceNet en cas de défaillance d'un esclave vers l'application. Reportez-vous au manuel d'utilisation de l'automate pour consulter leur description. Ces informations doivent être gérées dans l'application de l'automate.
<b>Réponse du scanner DeviceNet</b>
Le scanner DeviceNet fournit au processeur différents objets d'erreur et de diagnostic en cas de défaillance d'un esclave DeviceNet.
<b>Réponse de la passerelle LUF9</b>
En cas de défaillance, la passerelle cesse de communiquer avec le scanner DeviceNet et les esclaves Modbus.
<b>Réponse du Tesys U</b>
Selon la configuration du Tesys U : Si les démarreurs-contrôleurs ne reçoivent aucune requête, ils : stoppent le moteur, conservent le même état ou actionnent le moteur. Reportez-vous aux manuels d'utilisation du Tesys U pour régler ces positions de repli.

### Passerelles LUF9 déconnectées du côté Modbus ou défaillance du Tesys U

<b>Réponse de l'automate</b>
Le processeur donne accès au mot d'état de la passerelle provenant de la table d'entrée du scanner DeviceNet, ainsi qu'au mot de commande de la passerelle provenant de la table de sortie. Ces 2 mots doivent être gérés dans l'application de l'automate afin de détecter si un esclave Modbus est manquant.
<b>Réponse du scanner DeviceNet</b>
Le scanner DeviceNet doit être configuré de façon à accéder à l'état de la passerelle et aux mots de commande afin de fournir des informations de diagnostic Modbus.
<b>Réponse de la passerelle LUF9</b>
Avec la configuration par défaut de la passerelle : Timeout time = 300 ms, Retries = 3, Reconnect time = 10 sec et Offline option for sub-network = Clear. Après l'envoi d'une requête à un esclave, si aucune réponse ne parvient après 300 ms, la passerelle l'envoi de nouveau trois fois avant de fournir des informations relatives à l'esclave manquant dans le mot d'état de la passerelle. Toutes les données envoyées au scanner DeviceNet (requêtes de lecture) ont la valeur 0. La passerelle essaie de reconnecter l'esclave manquant en respectant le même ordre toutes les 10 secondes.
<b>Réponse du Tesys U</b>
Si la passerelle LUF9 est déconnectée du côté Modbus : Les démarreurs-contrôleurs ne reçoivent aucune requête. Selon leur configuration, ils : stoppent le moteur, conservent le même état ou actionnent le moteur. Reportez-vous aux manuels d'utilisation du Tesys U pour régler la position de repli.
En cas de défaillance du Tesys U : Aucune réponse n'est envoyée à la passerelle. L'état du moteur est indéterminé. Ce cas doit être géré dans l'application de l'automate.

# 4. Mise en œuvre logicielle de la passerelle

## 4.2. Configuration de la passerelle sous RSNetWorx

L'automate maître DeviceNet doit être configuré pour qu'il puisse avoir accès à l'ensemble des données décrites dans les chapitres Annexe B : Configuration par défaut, mémoire de données d'entrée et de sortie.

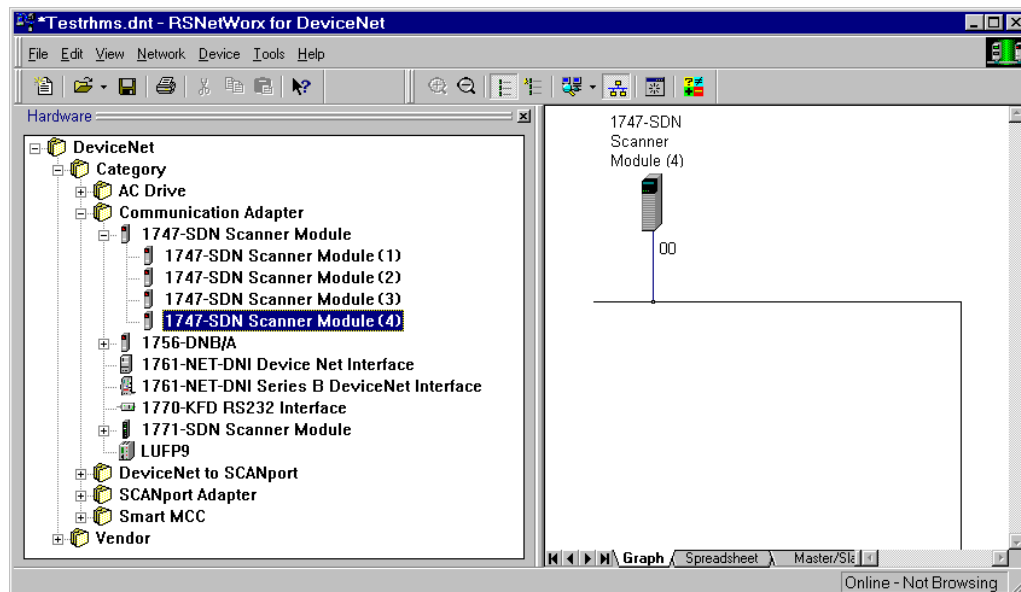
Les chapitres suivants décrivent les étapes de configuration sous RSNetWorx qu'il est nécessaire d'effectuer pour que la passerelle soit correctement reconnue par l'automate maître DeviceNet.

**REMARQUE** : Le réseau DeviceNet qui est décrit dans les chapitres suivants comporte un maître et un seul esclave (passerelle LUF9). Vous devrez donc adapter l'adressage des entrées et des sorties présenté ci-après (%IW et %QW) en fonction des autres esclaves du réseau DeviceNet que vous aurez à configurer.

### 4.2.1. Sélection et ajout du scanner DeviceNet de l'automate maître

Sous RSNetWorx, sélectionnez le type de scanner dont vous disposez et ajoutez-le à la topologie du réseau DeviceNet.

Dans notre exemple, ce scanner est un « 1747-SDN Scanner Module (4) » et son adresse MAC ID est égale à 00.



### 4.2.2. Mise en place du fichier de description de la passerelle

Le fichier EDS qui décrit la passerelle doit être placé sur le disque dur du PC afin que le logiciel RSNetWorx puisse y avoir accès à tout moment.

Ce fichier est disponible sur le site Web <http://www.schneider-electric.com>: "LUF9\_136.eds".

→ Une fois sous RSNetWorx, reportez-vous à sa documentation afin de prendre connaissance de la procédure à effectuer pour importer un fichier EDS. Cette procédure doit être ensuite appliquée au fichier « LUF9\_136.eds ». Elle utilise l'assistant « EDS wizard », accessible depuis le menu « Tools ».

Les deux entrées suivantes sont alors ajoutées dans l'arborescence des produits DeviceNet reconnus :

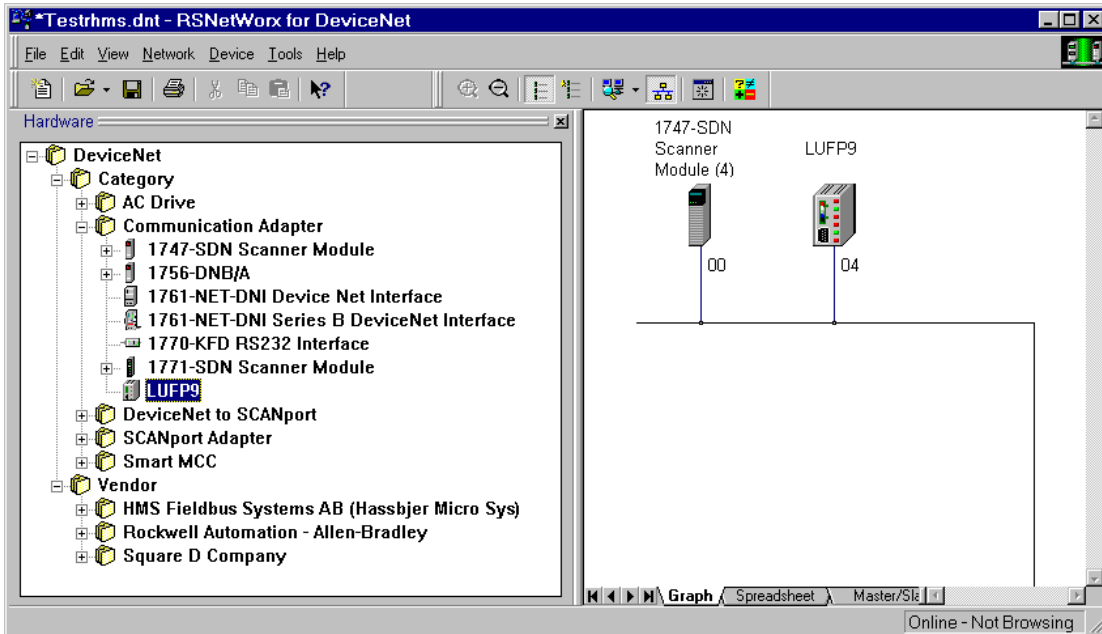
- DeviceNet / Category / Communication Adapter / LUF9
- DeviceNet / Vendor / Schneider Automation / LUF9

## 4. Mise en œuvre logicielle de la passerelle

### 4.2.3. Sélection et ajout d'une passerelle au réseau DeviceNet

Sélectionnez « LUFFP9 » dans la liste de gauche, puis ajoutez-le à la topologie du réseau DeviceNet.

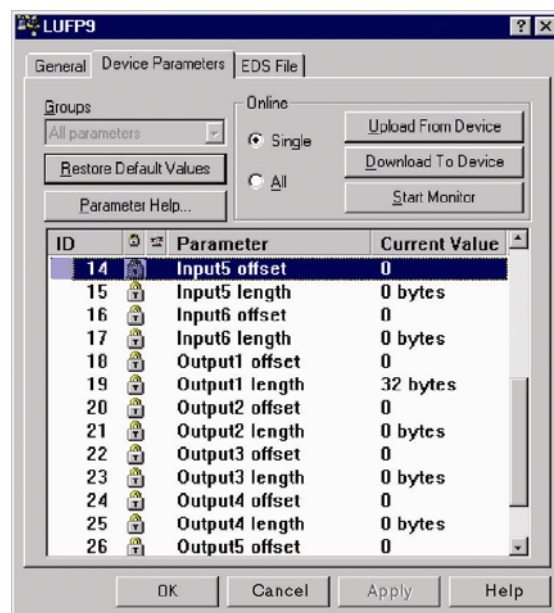
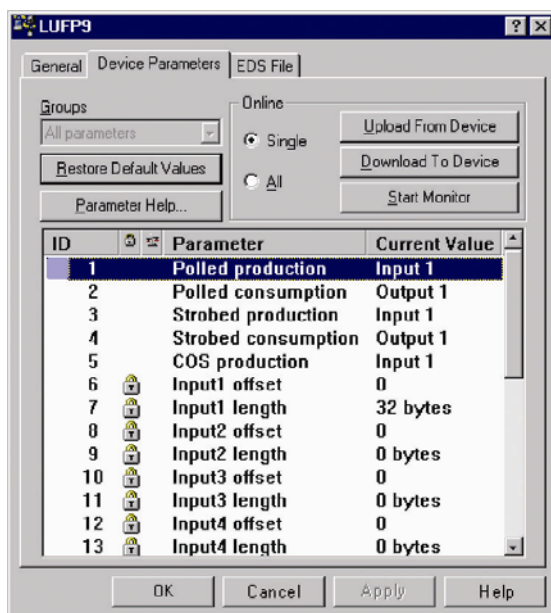
Dans notre exemple, nous avons affecté l'adresse Mac ID 04 à la passerelle (la configuration de l'adresse d'une passerelle est décrite dans le chapitre 2.7.2).



### 4.2.4. Modification des paramètres de la passerelle

Double-cliquez sur l'icône qui correspond à la passerelle, dans le cadre droit.

Dans la fenêtre qui apparaît alors, sélectionnez l'onglet « Device Parameters » et vérifiez que les valeurs des paramètres correspondent à celles des paramètres reproduits ci-dessous. Au besoin, modifiez-les (seuls les paramètres 1 à 5 sont accessibles en écriture pour l'utilisateur), puis cliquez sur le bouton « Download To Device » afin de transmettre ces modifications à la passerelle.





## 4. Mise en œuvre logicielle de la passerelle

---

En cas de doute sur l'affichage obtenu, cliquez sur le bouton « Upload From Device », puis sur « Start Monitor ». Le logiciel RSNetWorx commence alors à lire dans la passerelle les valeurs des paramètres affichés. Cliquez sur le bouton « Stop Monitor » pour arrêter ce processus de lecture.

Les paramètres les plus importants, dans le cas de la configuration par défaut de la passerelle, sont les paramètres 1 et 2 (transferts périodiques entre l'automate et la passerelle via une connexion périodique appelée « polled »), 6 et 7 (offset et taille de la zone des données d'entrée dans la mémoire d'entrée de la passerelle), 18 et 19 (offset et taille de la zone des données de sortie dans la mémoire de sortie de la passerelle). La valeur de chaque paramètre de type « offset » fait référence à un décalage depuis le début de la zone mémoire des données d'entrée de la passerelle.

**REMARQUE** : Seul le contrôle des zones « Input1 » et « Output1 » est évoqué dans ce guide. Le contrôle des zones Input2 à Input6 et Output2 à Output6 est une application avancée et ne constitue pas l'objet de ce guide. Contactez le service de support de *Schneider Electric* pour obtenir de l'aide concernant le contrôle de ces paramètres.

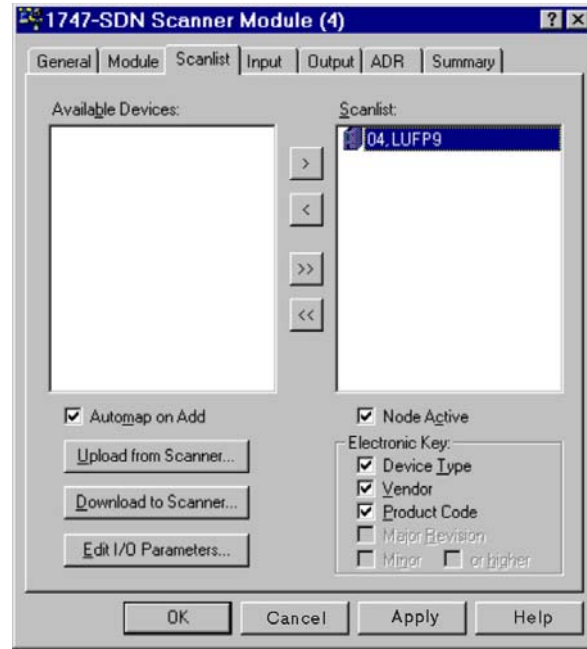
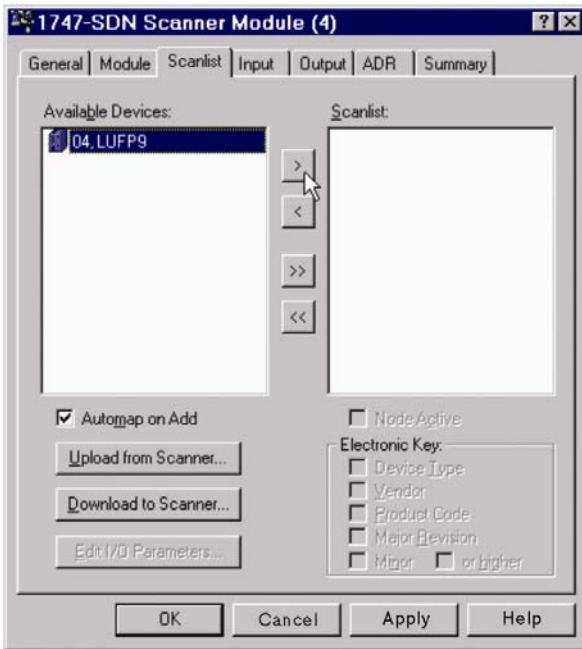
**REMARQUE** : Si vous créez ou modifiez une configuration à l'aide de ABC-LUFP Config Tool (voir le chapitre 6), vérifiez que les zones de données d'entrée / sortie définies dans la mémoire de la passerelle sont appropriées pour la nouvelle configuration et pour les communications à l'aide du maître DeviceNet. Ces zones de données d'entrée / sortie définissent l'ensemble des octets échangés avec les esclaves Modbus via les champs « Data » ou « Preset Data » (ou « Variable Data » pour les données de taille variable) des trames Modbus. Le non-respect de ces étapes peut générer une erreur de configuration.

## 4. Mise en œuvre logicielle de la passerelle

### 4.2.5. Configuration du Scanner DeviceNet

Double-cliquez sur l'icône qui correspond au scanner DeviceNet.

La fenêtre qui apparaît alors permet de configurer les échanges effectués par le scanner. Sélectionnez l'onglet « Scanlist » et ajoutez la passerelle « LUF9 » à la « Scanlist » (boutons  ou  ). Après sélection de la passerelle dans cette liste, le bouton « Edit I/O Parameters... » devient accessible.

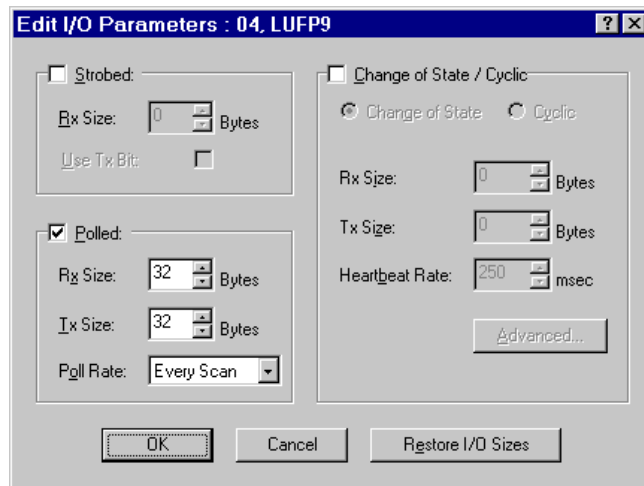


Cliquez sur le bouton « Edit I/O Parameters... ».

Dans la fenêtre qui apparaît alors, cochez la case « Polled: », puis configurez la taille des données reçues (Rx = 32 octets) et la taille des données émises (Tx = 32 octets) par le scanner.

Dans le cas de la configuration par défaut de la passerelle LUF9, ces valeurs permettent d'échanger l'ensemble des données présentées dans Annexe B : Configuration par défaut.

**REMARQUE :** Si vous créez ou modifiez une configuration à l'aide de ABC-LUF9 Config Tool, reportez-vous au chapitre 6.



### AVERTISSEMENT

#### RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

Etant donné que seules les zones « Input1 » et « Output1 » sont décrites dans ce guide et qu'il est interdit d'utiliser plus d'une connexion de type « I/O connection » dans la même zone, *vous ne devez pas configurer plus d'une « I/O connection » : par exemple, si vous configurez une connexion « Polled », vous ne devez pas configurer une connexion « Strobed » ou « Change of State / Cyclic ».*

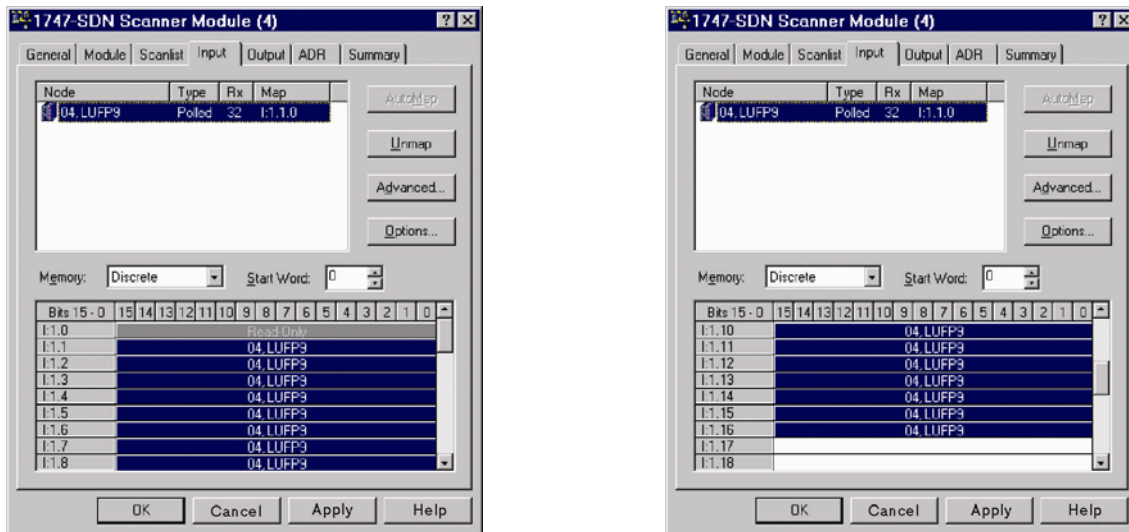
**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

## 4. Mise en œuvre logicielle de la passerelle

### 4.2.6. Configuration des entrées issues de la passerelle

Dans l'onglet « Input », sélectionnez la passerelle « LUFF9 », puis cliquez sur le bouton « AutoMap ». RSNetWorx établit alors de manière automatique la correspondance entre les 32 octets de données (format 8 bits) issues de la passerelle et les 16 entrées automate « I:1.1 » à « I:1.16 » (format 16 bits) correspondantes.

Vérifiez qu'une correspondance entre toutes les données issues de la passerelle et les entrées automate « I:1.1 » à « I:1.16 » a été établie.



La correspondance entre le contenu de la mémoire d'entrée de la passerelle (voir Annexe B : Configuration par défaut) et les entrées de l'automate « I:1.1 » à « I:1.16 » est donnée dans le tableau suivant :

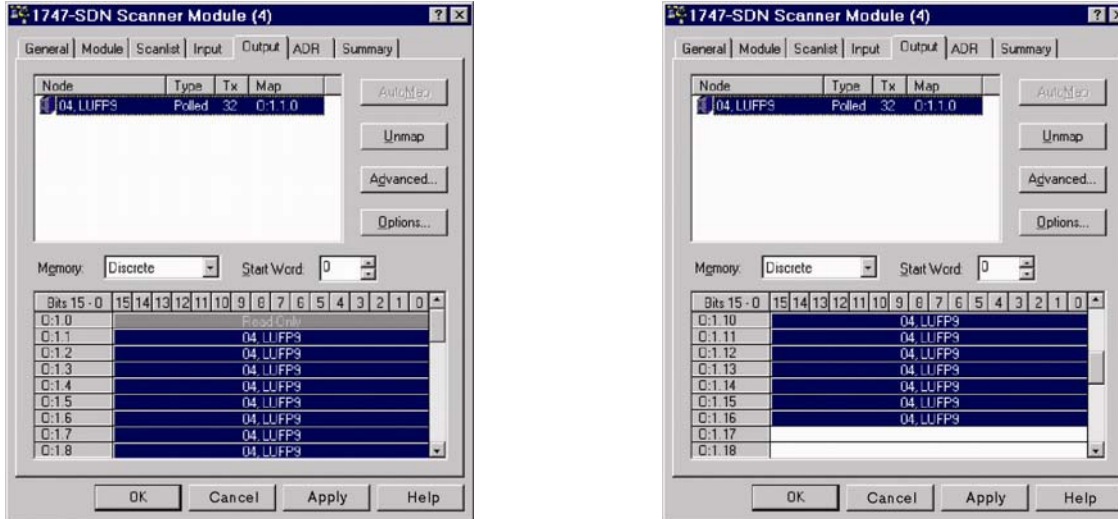
Service	Entrée Automate	Description	
		Bit 0.....Bit 7	Bit 8 .....Bit 15
Gestion du réseau aval Modbus (Mots d'état)	I:1.1	Mot d'état de la passerelle LUFF9 (MSB → 0xxx••)   (LSB → 0x••xx)	
Communications périodiques — Surveillance des départs-moteurs TeSys U	I:1.2	Valeur du registre d'état du départ-moteur ①	
	I:1.3	Valeur du registre d'état du départ-moteur ②	
	I:1.4	Valeur du registre d'état du départ-moteur ③	
	I:1.5	Valeur du registre d'état du départ-moteur ④	
	I:1.6	Valeur du registre d'état du départ-moteur ⑤	
	I:1.7	Valeur du registre d'état du départ-moteur ⑥	
	I:1.8	Valeur du registre d'état du départ-moteur ⑦	
Communications apériodiques — Lecture de la valeur d'un paramètre de départ-moteur (REPONSE)	I:1.10	Emplacement mémoire libre	N° esclave (0x01-0x08)
	I:1.11	Numéro de la fonction (0x03)	Nombre d'octets lus (0x02)
	I:1.12	Valeur du paramètre lu (MSB → 0xxx••)   (LSB → 0x••xx)	
Communications apériodiques — Ecriture de la valeur d'un paramètre de départ-moteur (REPONSE)	I:1.13	N° esclave (0x01-0x08)	N° fonction (0x06)
	I:1.14	Adresse du paramètre écrit (MSB → 0xxx••)   (LSB → 0x••xx)	
	I:1.15	Valeur du paramètre écrit (MSB → 0xxx••)   (LSB → 0x••xx)	
Communications apériodiques (« Trigger bytes » des réponses)	I:1.16	Compteur de réponse de la lecture d'un paramètre	Compteur de réponse de l'écriture d'un paramètre

## 4. Mise en œuvre logicielle de la passerelle

### 4.2.7. Configuration des sorties destinées à la passerelle

Dans l'onglet « Output », sélectionnez la passerelle « LUFF9 », puis cliquez sur le bouton « AutoMap ». RSNetWorx établit alors de manière automatique la correspondance entre les 32 octets de données (format 8 bits) à transmettre à la passerelle et les 16 sorties automate « O:1.1 » à « O:1.16 » (format 16 bits) correspondantes.

Vérifiez qu'une correspondance entre toutes les données destinées à la passerelle et les sorties automate « O:1.1 » à « O:1.16 » a été établie.



La correspondance entre le contenu de la mémoire de sortie de la passerelle (voir Annexe B : Configuration par défaut, zone mémoire des données de sortie) et les sorties de l'automate « O:1.1 » à « O:1.16 » est donnée dans le tableau suivant :

Service	Sortie Automate	Description	
		Bit 0 ..... Bit 7	Bit 8 ..... Bit 15
Gestion du réseau aval Modbus (Mot de commande)	O:1.1	Mot de commande du maître DeviceNet (MSB → 0xxx••)	(LSB → 0x••xx)
Communications périodiques — Commande des départs-moteurs TeSys U	O:1.2	Valeur du registre de commande du départ-moteur ①	
	O:1.3	Valeur du registre de commande du départ-moteur ②	
	O:1.4	Valeur du registre de commande du départ-moteur ③	
	O:1.5	Valeur du registre de commande du départ-moteur ④	
	O:1.6	Valeur du registre de commande du départ-moteur ⑤	
	O:1.7	Valeur du registre de commande du départ-moteur ⑥	
	O:1.8	Valeur du registre de commande du départ-moteur ⑦	
Communications aperiodiques — Lecture de la valeur d'un paramètre de départ-moteur (REQUETE)	O:1.9	Valeur du registre de commande du départ-moteur ⑧	
	O:1.10	N° esclave (0x01-0x08)	N° fonction (0x03)
	O:1.11	Adresse du paramètre à lire (MSB → 0xxx••)	(LSB → 0x••xx)
Communications aperiodiques — Ecriture de la valeur d'un paramètre de départ-moteur (REQUETE)	O:1.12	Nombre de paramètres à lire (MSB → 0x00••)	(LSB → 0x••01)
	O:1.13	N° esclave (0x01-0x08)	N° fonction (0x06)
	O:1.14	Adresse du paramètre à écrire (MSB → 0xxx••)	(LSB → 0x••xx)
Communications aperiodiques — Ecriture de la valeur d'un paramètre de départ-moteur (REQUETE)	O:1.15	Valeur du paramètre à écrire (MSB → 0xxx••)	(LSB → 0x••xx)
	O:1.16	Compteur de requête de la lecture d'un paramètre	Compteur de requête de l'écriture d'un paramètre

### 4.2.8. Transfert de la configuration du scanner DeviceNet

## 4. Mise en œuvre logicielle de la passerelle

Une fois que vous aurez achevé les opérations précédemment décrites, assurez-vous que les modifications effectuées sont transmises au scanner DeviceNet. Pour cela, cliquez sur le bouton « Download to Scanner... » présent dans chacun des onglets « Module » et « Scanlist » de la fenêtre des propriétés du scanner DeviceNet. Si besoin est, reportez-vous à la documentation de RSNetWorx pour de plus amples détails à ce sujet.

### 4.2.9. Développement d'une application DeviceNet

L'automate maître DeviceNet pris pour exemple est un SLC500, commercialisé par *Allen Bradley*. Un exemple d'application automate, développé sous RSLogix 500, est présenté dans Annexe C : Exemple d'utilisation (RSLogix 500)

Cet exemple utilise l'automate, la passerelle et les 8 départs-moteurs TeSys U présentés dans Mise en œuvre logicielle de la passerelle.

### 4.3. Description des services affectés aux entrées/sorties de la passerelle

**Gestion du réseau aval Modbus :** Reportez-vous au chapitre 5.2, pour obtenir une description détaillée de ce service. L'exemple fourni et décrit dans Annexe C : Programme principal, effectue uniquement un acquittement automatique des diagnostics de la passerelle, c'est-à-dire qu'il n'exploite pas les données de ces diagnostics. Dans le cas de la configuration par défaut de la passerelle, dans ABC-LUFP Config Tool, le champ « Control/Status Byte » de l'élément « ABC-LUFP » est défini sur « Enabled but no startup lock ».

**Communications périodiques (entrées) :** La valeur de chacun des 8 mots de ce service correspond à celle du registre d'état d'un départ-moteur TeSys U (registre situé à l'adresse 455).

**Communications périodiques (sorties) :** La valeur de chacun des 8 mots de ce service correspond à la valeur à destination du registre de commande d'un départ-moteur TeSys U (registre situé à l'adresse 704).

Reportez-vous à l'Annexe C Sous-programme de commande/surveillance, pour consulter un exemple d'utilisation simplifiée de ces services de « communications périodiques ».

**Communications apériodiques :** Reportez-vous à l'Annexe C : Sous-programme de lecture d'un paramètre... et Sous-programme d'écriture d'un paramètre..., pour consulter un exemple d'utilisation des services de « communications apériodiques ».

Ces services de communications apériodiques proposent des fonctions similaires à celles des « variables périodiques indexées », ou PKW, que l'on peut trouver sur certains produits *Schneider Electric*, tels que les variateurs de vitesse de type ATV.

Lors de l'utilisation d'entrées et de sorties 16 bits pour lesquelles l'ordre du LSB et du MSB est spécifié, le maître DeviceNet utilise l'arrangement d'octets Big Endian (LSB MSB), alors que les esclaves Modbus utilisent l'arrangement d'octets Little Endian (MSB LSB). Dans de nombreuses situations, le maître DeviceNet gère cette conversion en interne. Il est possible que cela ne soit pas le cas avec certaines configurations, certains services apériodiques ou certaines applications personnalisées. Il est nécessaire que ce comportement soit correctement identifié avant de mettre le système en service.

### AVERTISSEMENT

#### RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

L'utilisateur doit s'assurer que la conversion des arrangements Endian (arrangement des octets dans un mot de 16 bits) est correcte entre les bus de terrain DeviceNet et Modbus. Durant la configuration du maître DeviceNet, lors de l'utilisation d'applications personnalisées ou lors d'une opération de programmation visant à établir une communication entre le maître DeviceNet et les esclaves Modbus par l'intermédiaire de la passerelle, la gestion des arrangements Endian (arrangement des octets dans un mot de 16 bits) doit être correcte pour chaque bus de terrain. Si l'ordre des octets transmis à des entrées et à des sorties de 16 bits est géré de façon inappropriée, des données incorrectes peuvent être écrites dans la configuration du périphérique Modbus ou dans les registres de commande, provoquant un fonctionnement imprévu de l'appareil.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

## 4. Mise en œuvre logicielle de la passerelle

- **Exemple de lecture d'un paramètre de départ-moteur :**

Lecture du 1er registre de défaut (adresse = 452 = 0x01C4) sur le départ-moteur « TeSys U n°5 ». Les valeurs initiales de O:1.16 et de I:1.16 sont égales à 0x1306. Le résultat de la lecture est 0x0002 (défaut magnétique).

Sortie	Valeur	Signification (MSB + LSB)
O:1.10	0x0305	N° Fonction + N° Esclave
O:1.11	0xC401	Adresse paramètre (MSB↔LSB)
O:1.12	0x0100	Nombre de paramètres (MSB↔LSB)
O:1.16	0x1307	« Trigger byte » de la requête (Pf)

Entrée	Valeur	Signification (MSB + LSB)
I:1.10	0x0500	N° Esclave + (non utilisé)
I:1.11	0x0203	Nombre d'octets + N° Fonction
I:1.12	0x0200	Valeur lue (MSB↔LSB)
I:1.16	0x1307	« Trigger byte » de la réponse (Pf)

- **Exemple d'écriture d'un paramètre de départ-moteur :**

Écriture du 2nd registre de commande (adresse = 705 = 0x02C1) sur le départ-moteur TeSys U n°7 à la valeur 0x0006 (RAZ statistiques + RAZ mémoire thermique). Les valeurs initiales de O:1.16 et de I:1.16 sont égales à 0x1307. Le résultat de l'écriture est un écho à la commande, c'est-à-dire que les valeurs des champs « adresse paramètre » et « valeur à écrire » sont identiques dans la requête et dans la réponse.

Sortie	Valeur	Signification (MSB + LSB)
O:1.13	0x0607	N° Fonction + N° Esclave
O:1.14	0xC102	Adresse paramètre (MSB↔LSB)
O:1.15	0x0600	Valeur à écrire (MSB↔LSB)
O:1.16	0x1407	« Trigger byte » de la requête (PF)

Entrée	Valeur	Signification (MSB + LSB)
I:1.13	0x0607	N° Fonction + N° Esclave
I:1.14	0xC102	Adresse paramètre (MSB↔LSB)
I:1.15	0x0600	Valeur à écrire (MSB↔LSB)
I:1.16	0x1407	« Trigger byte » de la réponse (PF)

Aucune vérification des erreurs n'est effectuée pour les données transmises à l'aide des services aperiodiques décrits ci-dessus. Les valeurs incorrectes écrites vers les sorties et qui correspondent à des services de communication aperiodiques provoqueront la transmission d'un cadre Modbus incohérent. Ce cadre Modbus incohérent peut retourner une erreur ou provoquer un comportement imprévisible des périphériques esclaves.

### AVERTISSEMENT

#### RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

L'utilisateur doit vérifier les erreurs et les gérer de façon appropriée pour les valeurs écrites vers les sorties, qui correspondent aux services de communication aperiodiques. L'envoi de valeurs incorrectes aux sorties des services aperiodiques peut provoquer un comportement imprévisible du système.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

# 5. Initialisation et diagnostic de la passerelle

Ce chapitre décrit le principe de l'initialisation et du diagnostic de la passerelle selon chacune des trois options offertes par celle-ci. Ces options peuvent être configurées via ABC-LUFP Config Tool, en modifiant l'affectation du champ « Control/Status Byte » de l'élément « ABC-LUFP » (voir chapitre 6.13.2). Ces options sont les suivantes :

<u>Champ « Control/Status Byte » :</u>	<u>Signification :</u>
Activé .....	Gestion complète
Enabled but no startup lock .....	Diagnostic et commande
Désactivé .....	Fonctionnement simplifié

*L'option retenue dans le cas de la configuration par défaut est l'option « Enabled but no startup lock », encadrée ci-dessus.*

<b>Gestion complète</b>	Gestion dans l'application d'automate : → du démarrage des échanges Modbus cycliques → de l'activation / désactivation des esclaves Modbus → du diagnostic du réseau Modbus.
<b>Diagnostic et commande</b>	Gestion dans l'application d'automate : → de l'activation / désactivation des esclaves Modbus → du diagnostic du réseau Modbus.
<b>Fonctionnement simplifié</b>	→ Démarrage automatique des échanges Modbus cycliques → Aucune activation / désactivation des esclaves Modbus → Aucun diagnostic du réseau Modbus

## 5.1. Gestion complète

Le maître DeviceNet gère le démarrage des échanges Modbus cycliques, l'activation et la désactivation des esclaves Modbus ainsi que le diagnostic Modbus au moyen de 2 mots :

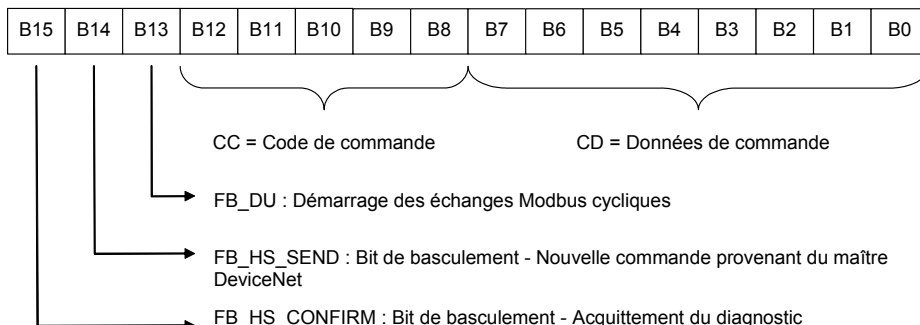
- Un mot de commande DeviceNet transmis par l'application de l'automate et associé aux adresses 0x0200 et 0x0201 de la mémoire de sortie de la passerelle ;
- Un mot d'état de la passerelle transmis par la passerelle et associé aux adresses 0x0000 et 0x0001 de la mémoire d'entrée de la passerelle.

Le mot d'état de la passerelle n'est pas actualisé de façon cyclique. La mise à jour de ce mot repose sur un système de bit de basculement qui doit être géré dans l'application de l'automate :

Le diagnostic est actualisé par la passerelle à l'aide du bit de basculement B15.

Toute nouvelle commande provenant du maître DeviceNet est envoyée à l'aide du bit de basculement B14.

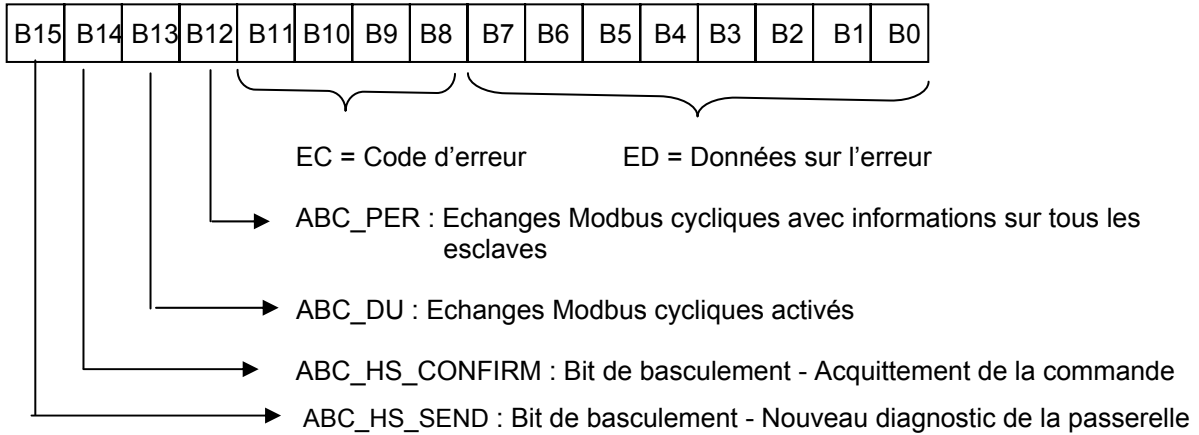
### 5.1.1. Mot de commande du maître DeviceNet



# 5. Initialisation et Diagnostic de la passerelle

Reportez-vous au chapitre 5.4 pour consulter la description détaillée de chaque bit.

## 5.1.2. Mot d'état de la passerelle



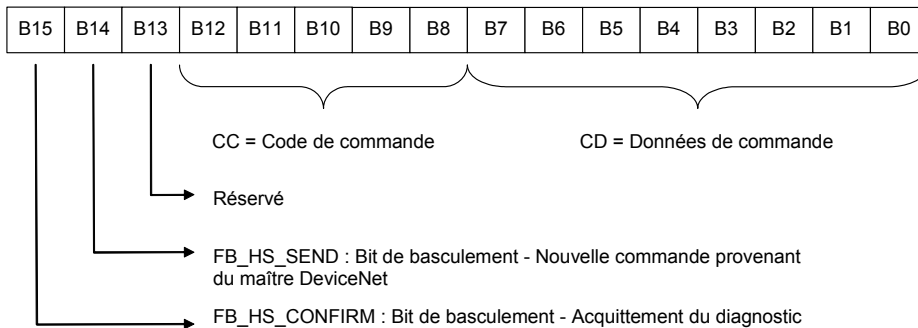
Reportez-vous au chapitre 5.5 pour consulter la description détaillée de chaque bit.

## 5.2. Diagnostic et commande

Le maître DeviceNet gère l'activation et la désactivation des esclaves Modbus ainsi que le diagnostic du réseau Modbus à l'aide des 2 mêmes mots que pour la gestion complète.

Les bits de gestion des échanges Modbus cycliques sont inactifs.

### 5.2.1. Mot de commande du maître DeviceNet

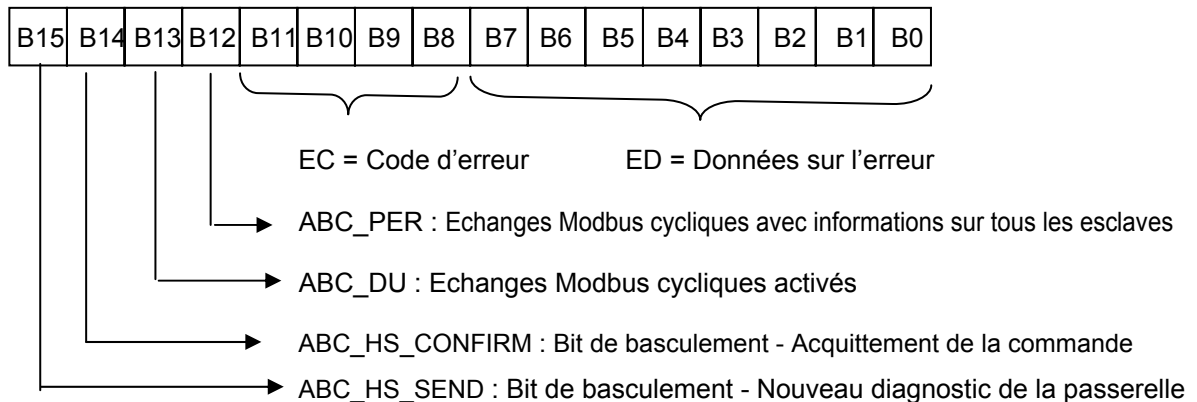


Reportez-vous au chapitre 5.4 pour consulter la description détaillée de chaque bit.



# 5. Initialisation et diagnostic de la passerelle

## 5.2.2. Mot d'état de la passerelle



Reportez-vous au chapitre 5.5 pour consulter la description détaillée de chaque bit.

Dans les modes « Gestion complète » et « Diagnostic et commande », il est important que vous configurez votre maître DeviceNet pour qu'il ait accès aux deux premiers octets de la zone des données de sortie de la passerelle, ainsi qu'aux deux premiers octets de la zone des données d'entrée de la passerelle.

### **⚠ AVERTISSEMENT**

#### **CONFIGURATION ERRONEE DES ZONES DE DONNEES DE LA PASSERELLE LUPP•**

Configurez votre maître DeviceNet pour qu'il ait accès aux deux premiers octets de la zone des données de sortie de la passerelle, ainsi qu'aux deux premiers octets de la zone des données d'entrée de la passerelle. Un défaut de configuration de l'accès à ces octets peut engendrer une incapacité à arrêter les communications Modbus et empêcher l'enregistrement des conditions d'erreur pour une évaluation ultérieure. Chacune de ces conséquences peut provoquer un fonctionnement imprévu de l'appareil.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

Pour plus d'informations, reportez-vous au chapitre 4.2.

## 5.3. Fonctionnement simplifié

Les deux registres de 16 bits situés aux adresses 0x0000-0x0001 (entrées) et 0x0200-0x0201 (sorties) ne sont plus utilisés. Ces deux adresses peuvent donc être utilisées pour échanger des données avec l'esclave Modbus.

Aucun diagnostic n'est renvoyé à l'automate. Le mot de commande du maître DeviceNet et le mot d'état de la passerelle n'existent pas durant des opérations simplifiées.

## 5. Initialisation et Diagnostic de la passerelle

### 5.4. Description du mot de commande du maître DeviceNet

Le mot de sortie situé aux adresses 0x0200 (MSB) et 0x0201 (LSB) dans la mémoire de sortie de la passerelle constitue le mot de commande du maître DeviceNet. Sa structure est décrite ci-dessous :

Bits	Description
15	<p><b>FB_HS_CONFIRM : Bit d'acquiescement d'un diagnostic de la passerelle</b></p> <p>Le maître DeviceNet doit comparer la valeur du bit FB_HS_CONFIRM à celle du bit ABC_HS_SEND (bit 15 du mot d'état de la passerelle). Si ces deux valeurs sont différentes, cela signifie que la passerelle a transmis un nouveau diagnostic au maître DeviceNet.</p> <p>Pour indiquer à la passerelle qu'il a pris connaissance d'un diagnostic, le maître DeviceNet doit recopier la valeur du bit ABC_HS_SEND dans le bit FB_HS_CONFIRM. Cela autorise la passerelle à émettre un nouveau diagnostic.</p> <p><b>Récapitulatif :</b></p> <ul style="list-style-type: none"><li>• Si ( FB_HS_CONFIRM = ABC_HS_SEND ) → Le mot d'état de la passerelle contient un diagnostic qui a déjà été acquiescé par le maître DeviceNet. La passerelle pourra donc à nouveau utiliser ce mot d'état pour y placer un autre diagnostic.</li><li>• Sinon → Un nouveau diagnostic est disponible dans le mot d'état de la passerelle. Le maître DeviceNet peut lire ce diagnostic, mais doit également recopier la valeur de ABC_HS_SEND dans FB_HS_CONFIRM afin d'autoriser la passerelle à générer de nouveaux diagnostics.</li></ul>
14	<p><b>FB_HS_SEND : Bit de basculement - Nouvelle commande provenant du maître DeviceNet</b></p> <p>Avant de modifier la valeur de FB_DU, le maître DeviceNet doit comparer les valeurs de FB_HS_SEND et de ABC_HS_CONFIRM (bit 14 du mot d'état de la passerelle). Si ces deux valeurs sont différentes, cela signifie que la passerelle n'a pas encore tenu compte de la commande précédente du maître DeviceNet. Dans le cas contraire, le maître DeviceNet peut transmettre une nouvelle commande. Il peut alors mettre à jour le bit FB_DU en fonction de la nature de sa commande (arrêt ou activation des échanges Modbus), puis faire basculer la valeur du bit FB_HS_SEND afin de signifier à la passerelle qu'il lui a transmis une nouvelle commande.</p> <p><b>Récapitulatif :</b></p> <ul style="list-style-type: none"><li>• Si ( FB_HS_SEND ≠ ABC_HS_CONFIRM ) → Le mot de commande du maître DeviceNet contient une commande n'ayant pas encore été acquiescée par la passerelle. Le maître DeviceNet ne peut donc pas utiliser ce mot pour y placer une nouvelle commande.</li><li>• Sinon → La commande précédente du maître DeviceNet a été acquiescée par la passerelle, ce qui l'autorise à émettre une nouvelle commande. Dans ce cas, il modifie la valeur du bit FB_DU, puis fait basculer la valeur du bit FB_HS_SEND.</li></ul>
13	<p><b>FB_DU : Mise en route des échanges Modbus</b></p> <p><i>(Réservé en mode « Diagnostic et commande »)</i></p> <p>La mise à un de ce bit par le maître DeviceNet sert à autoriser les communications entre la passerelle et les esclaves Modbus. Sa remise à zéro sert à les inhiber.</p> <p>Lorsque le maître DeviceNet met ce bit à un, il est préférable que l'ensemble des données de sortie qu'il aura placées dans la mémoire de sortie de la passerelle soient à jour (« FB_DU » signifie « FieldBus – Data Updated »). Si elles ne le sont pas, ces données seront transmises telles quelles aux esclaves Modbus.</p> <p><b>REMARQUE :</b> Tant que le maître DeviceNet ne met pas le bit FB_DU à 1, la passerelle n'envoie aucune requête aux esclaves Modbus. Ce bit est principalement utilisé par un maître DeviceNet afin d'empêcher la passerelle d'envoyer des données invalides aux esclaves.</p>
8-12	<p><b>CC : Code de commande d'activation et de désactivation des esclaves Modbus</b></p> <p>Code de commande envoyé par le maître DeviceNet à la passerelle afin d'activer ou inhiber les communications avec un ou plusieurs esclaves Modbus (voir le tableau CC-CD).</p>
0-7	<p><b>CD : Données de commande d'activation et de désactivation des esclaves Modbus</b></p> <p>Données associées au code de commande CC (voir le tableau CC-CD).</p>

## 5. Initialisation et diagnostic de la passerelle

En raison de l'inversion du LSB et du MSB de ce registre entre la passerelle et le maître DeviceNet, la structure du mot de sortie correspondant (« O:1.1 » dans le cas de la configuration par défaut) est la suivante :

Bits	Description
8-15	<b>CD : Données de commande d'activation et de désactivation des esclaves Modbus</b>
7	<b>FB_HS_CONFIRM : Bit d'acquiescement d'un diagnostic de la passerelle</b>
6	<b>FB_HS_SEND : Nouvelle commande du maître DeviceNet</b>
5	<b>FB_DU : Mise en route des échanges Modbus (Réservé en mode « Diagnostic et commande »)</b>
0-4	<b>CC : Code de commande d'activation et de désactivation des esclaves Modbus</b>

*Exemple* : Si le mot de sortie O:1.1 est égal à 0x00A0, le mot de commande du maître DeviceNet sera égal à 0xA000.

L'utilisation correcte de ce mot de commande par le maître DeviceNet, afin de transmettre une nouvelle commande à la passerelle, passe par les étapes suivantes :

- Vérification de ( FB\_HS\_SEND = ABC\_HS\_CONFIRM ) ; Si FB\_HS\_SEND = ABC\_HS\_CONFIRM, alors
- la commande de mise en route des échanges Modbus (FB\_DU) est mise à jour,
- la commande des esclaves Modbus (via CC et CD) est mise à jour si le maître souhaite inhiber/activer un ou plusieurs esclaves,
- la valeur du bit FB\_HS\_SEND est inversée.

**REMARQUE** : Il est possible de simplifier cette utilisation de la manière suivante :

- Mise à un des bits FB\_DU et FB\_HS\_SEND pour activer les communications Modbus.
- Remise à zéro des bits FB\_DU et FB\_HS\_SEND pour arrêter les communications Modbus.

Même si l'écriture de données à 8 bits et de données à 16 bits dans le mot de commande du maître DeviceNet est possible en théorie, l'écriture directe dans le mot de commande du maître DeviceNet de données au format 16 bits peut engendrer des erreurs. Ce type d'écriture de données à 16 bits peut perturber le fonctionnement du transfert des diagnostics de la passerelle (modification non souhaitée de FB\_HS\_CONFIRM).

### **AVERTISSEMENT**

#### **RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT**

N'écrivez pas directement au format 16 bits dans le mot de commande du maître DeviceNet, car cela peut perturber le fonctionnement du transfert des informations de diagnostic de la passerelle vers le maître. Selon la configuration de l'utilisateur, cela peut provoquer un fonctionnement imprévu de l'appareil.

Au lieu d'écrire au format 8 ou 16 bits, vous devriez écrire, bit à bit, dans le mot de commande du maître DeviceNet. Par exemple, pour mettre à jour le bit FB\_DU, vous devriez écrire uniquement la valeur du bit 5 (ex. : O:1.1/5 dans le cas d'une configuration par défaut) sans modifier les autres bits de ce mot.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

## 5. Initialisation et Diagnostic de la passerelle

Les valeurs des champs CC et CD sont décrites dans le tableau suivant :

CC	Description de la commande	CD	Remarques
2#00000	Aucune commande	—	—
2#10000	Désactiver un nœud	Adresse Modbus de l'esclave à désactiver	La passerelle inhibe <i>tous</i> les échanges Modbus (commandes <i>et</i> transactions) configurés pour l'esclave Modbus correspondant. <b>Remarque</b> : Dans le cas de la configuration par défaut de la passerelle LUF9, notez que la désactivation de l'esclave à l'adresse 1 ( <i>c'est-à-dire</i> : « TeSys U n°1 ») entraînera également l'inhibition des deux transactions destinées à lire/écrire n'importe quel paramètre de n'importe quel esclave.
2#10001	Activer un nœud (1)	Adresse Modbus de l'esclave à activer	La passerelle active <i>tous</i> les échanges Modbus (commandes <i>et</i> transactions) configurés pour l'esclave Modbus correspondant.
2#10010	Activer plusieurs nœuds (1)	Nombre d'esclave Modbus à activer	La passerelle active <i>tous</i> les échanges Modbus (commandes <i>et</i> transactions) configurés pour les premiers esclaves Modbus <b>CD</b> et inhibe <i>tous</i> les échanges Modbus configurés pour les esclaves Modbus restants. Si <b>CD</b> est supérieur ou égal au nombre total d'esclaves, tous les esclaves sont alors activés. <i>Exemple</i> : Dans le cas de la configuration par défaut, si <b>CD</b> = 5, les 5 premiers esclaves (« TeSys U n°1 » à « TeSys U n°5 ») sont alors activés tandis que les 3 esclaves restants (« TeSys U n°6 » à « TeSys U n°8 ») sont inhibés.

(1) Par défaut, tous les nœuds sont activés. Par conséquent, il n'est pas nécessaire d'activer un nœud qui n'a pas été préalablement désactivé.

## 5. Initialisation et diagnostic de la passerelle

### 5.5. Description du mot d'état de la passerelle

Le mot d'entrée situé aux adresses 0x0000 (MSB) et 0x0001 (LSB) dans la mémoire d'entrée de la passerelle constitue le mot d'état de la passerelle. Sa structure est décrite ci-dessous :

Bits	Description
15	<b>ABC_HS_SEND : Nouveau diagnostic de la passerelle</b> (Voir description du bit 15 du mot de commande du maître DeviceNet, FB_HS_CONFIRM.)
14	<b>ABC_HS_CONFIRM : Bit d'acquittement d'une commande du maître DeviceNet</b> (Voir description du bit 14 du mot de commande du maître DeviceNet, FB_HS_SEND.)
13	<b>ABC_DU : Echanges Modbus activés</b> La passerelle active ce bit pour signaler au maître DeviceNet que toutes les données Modbus qui sont situées dans sa zone mémoire d'entrée ont été mises à jour au moins une fois depuis la dernière activation de FB_DU (« ABC_DU » signifie « ABC – Data Updated »). Ces données Modbus d'entrée comprennent toutes les données des réponses de tous les esclaves Modbus, commandes périodiques et commandes aperiodiques confondues.  Ce bit est désactivé par la passerelle lorsque le bit FB_DU est désactivé, c'est-à-dire lorsque le maître DeviceNet demande l'arrêt des échanges Modbus. <b>REMARQUE</b> : Une fois qu'il est actif, ce bit n'est pas désactivé lorsque surviennent des erreurs de communication avec les esclaves Modbus. Pour signaler ce type d'erreurs, la passerelle utilise le bit 12 de son mot d'état.
12	<b>Périodicité des échanges Modbus</b> La passerelle active ce bit tant qu'elle communique de manière périodique avec l'ensemble des esclaves Modbus. Elle le désactive dès qu'elle perd la communication avec l'un d'entre eux.  Les éléments « Reconnect time (10ms) », « Retries » et « Timeout time (10ms) » de chacune des requêtes Modbus (voir chapitre 6.12.2.2) sont utilisés pour déterminer s'il y a perte, puis reprise, de communication. <b>REMARQUE</b> : Si plusieurs échanges périodiques sont configurés pour un même esclave Modbus, il suffit qu'un seul d'entre eux reste actif pour que les communications périodiques avec cet esclave soient déclarées actives.
8-11	<b>EC : Code de l'erreur associée au réseau Modbus</b> Code de l'erreur détectée sur le réseau Modbus par la passerelle et émise au maître DeviceNet (voir le tableau EC-ED).
0-07	<b>ED : Donnée de l'erreur associée au réseau Modbus</b> Donnée associée au code d'erreur EC (voir le tableau EC-ED).

En raison de l'inversion du LSB et du MSB de ce registre entre la passerelle et le maître DeviceNet, la structure du mot d'entrée correspondant (« I:1.1 » dans le cas de la configuration par défaut) est la suivante :

Bits	Description
8-15	<b>ED : Donnée de l'erreur associée au réseau Modbus</b>
7	<b>ABC_HS_SEND : Nouveau diagnostic de la passerelle</b>
6	<b>ABC_HS_CONFIRM : Bit d'acquittement d'une commande du maître DeviceNet</b>
5	<b>ABC_DU : Echanges Modbus activés</b>
4	<b>Périodicité des échanges Modbus</b>
0-3	<b>EC : Code de l'erreur associée au réseau Modbus</b>

*Exemple* : Si le mot d'état de la passerelle est égal à 0xF031, le mot d'entrée I:1.1 sera égal à 0x31F0.

## 5. Initialisation et Diagnostic de la passerelle

L'utilisation correcte de ce mot d'état par le maître DeviceNet, afin de prendre connaissance d'un diagnostic généré par la passerelle, passe par les étapes suivantes :

- Vérification de (ABC\_HS\_SEND ≠ FB\_HS\_CONFIRM) ; Si ABC\_HS\_SEND ≠ FB\_HS\_CONFIRM, alors
- la valeur de ABC\_DU est lue afin de déterminer si toutes les données Modbus d'entrée sont à jour ;
- la valeur du bit « Périodicité des échanges Modbus » est lue afin de déterminer si la périodicité des communications Modbus est maintenue ;
- les valeurs de EC et de ED sont lues afin de prendre connaissance d'une éventuelle erreur détectée par la passerelle sur le réseau Modbus (voir tableau ci-dessous) ;
- la valeur du bit ABC\_HS\_SEND est copiée dans le bit FB\_HS\_CONFIRM.

Cette dernière étape est très importante si le système est conçu pour lire les diagnostics de la passerelle et effectuer certaines actions en fonction du résultat. La copie de la valeur du bit ABC\_HS\_SEND dans le bit FB\_HS\_CONFIRM permet à la passerelle de transmettre un diagnostic futur, empêchant la perte d'informations relatives aux erreurs ultérieures.

### AVERTISSEMENT

#### RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

L'utilisateur doit s'assurer que la programmation du maître DeviceNet conclut des opérations de lecture en copiant la valeur du bit ABC\_HS\_SEND sur le bit FB\_HS\_CONFIRM. En cas d'omission de cette étape dans les applications dans lesquelles les diagnostics de la passerelle sont lus et génèrent une réaction, les informations de diagnostics futurs seront bloquées. Selon la configuration de l'utilisateur, cela peut provoquer un fonctionnement imprévu de l'appareil.

Par exemple, la disparition d'un esclave Modbus (EC = 2#0001) peut perturber les communications avec les autres esclaves, en raison des futures tentatives de reconnexion et des timeouts avec cet esclave Modbus défectueux. En conséquence et selon les besoins de votre application, il peut être très important pour le maître DeviceNet d'acquiescer chaque diagnostic afin d'être informé le plus rapidement possible de la disparition d'un esclave. Ainsi, votre application peut réagir en conséquence (*par exemple*, en inhibant l'esclave défectueux avec CC et CD du mot de commande de la passerelle).

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

Les valeurs des champs EC et ED sont décrites dans le tableau suivant :

CE	Description de l'erreur	ED	Remarques
2#0000	Ré-émissions sur le réseau Modbus	Nombre de ré-émissions	Nombre total de ré-émissions effectuées sur le sous-réseau, tous esclaves confondus.
2#0001	Absence d'un esclave Modbus	Adresse de l'esclave Modbus absent	—
2#0010	Absence de plusieurs esclaves Modbus	—	—
2#0011	Données excessives dans une réponse Modbus	Adresse de l'esclave Modbus incriminé	Cette erreur se produit lorsque la passerelle reçoit un trop grand nombre de données dans la réponse émise par l'un de ses esclaves Modbus.
2#0100	Erreur Modbus inconnue	Adresse de l'esclave Modbus incriminé	—
2#1111	Absence d'erreur	—	Code d'absence d'erreur utilisé par la passerelle lorsque aucune erreur de communication Modbus n'est détectée. Il est généralement utilisé lorsque des esclaves Modbus auparavant absents sont de nouveau sur le sous-réseau.

## 5. Initialisation et diagnostic de la passerelle

---

Le compteur de ré-émissions utilisé pour signaler cette erreur n'est pas remis à zéro lorsque la passerelle génère ce code d'erreur. En cas de problèmes récurrents de communication sur le réseau Modbus, la passerelle générera ce même diagnostic de manière répétitive, afin de renseigner le plus souvent possible le maître DeviceNet du nombre total de ré-émissions effectuées. La remise à zéro de ce compteur est effectuée lorsque sa valeur dépasse sa valeur maximale (compteur modulo 256 : 0xFF → 0x00).

En cas de déconnexion d'un ou de plusieurs équipements du sous-réseau Modbus, la passerelle LUF9 commence par reporter des erreurs de ré-émissions, puis l'erreur « Absence d'un esclave Modbus » ou « Absence de plusieurs esclaves Modbus ». Ensuite, lorsque la passerelle procède aux tentatives de re-connexion, seule l'erreur de ré-émission est reportée. De ce fait, l'indication des erreurs « Absence d'un esclave Modbus » ou « Absence de plusieurs esclaves Modbus » peut être perçue comme fugitive.

## 6. Configuration de la passerelle

Chacune des parties du chapitre présent décrit une étape distincte permettant de personnaliser la configuration de la passerelle, selon les besoins de l'utilisateur. Chaque partie présente une opération élémentaire en l'isolant du reste de la configuration et en décrivant les opérations à effectuer à l'aide de ABC-LUFP Config Tool (principalement) et de RSNetWorx (si besoin est), ainsi que leurs implications sur le comportement général de la passerelle.

Dans tous les cas, les deux premières étapes sont obligatoires, puisqu'elles permettent d'établir le dialogue entre la passerelle et le logiciel du PC qui permet de la configurer, c'est-à-dire ABC-LUFP Config Tool.

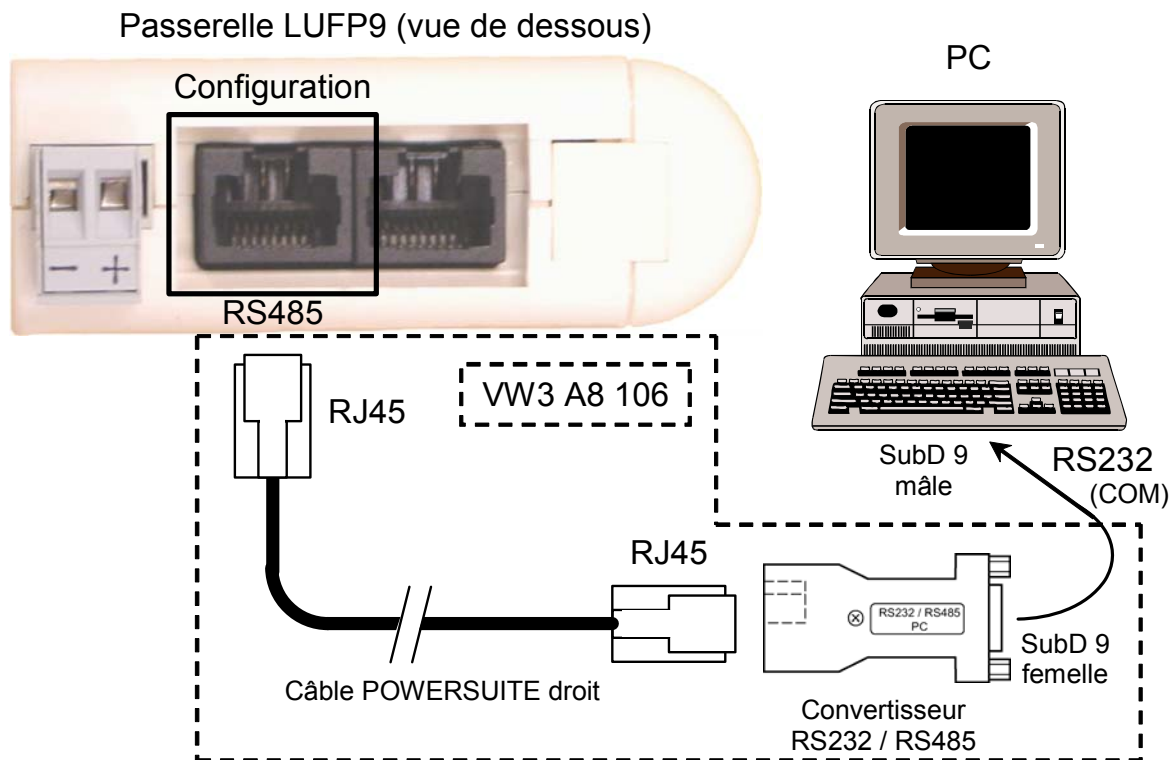
La lecture du chapitre 4 est fortement recommandée, car toutes les opérations effectuées sous ABC-LUFP Config Tool ou sous RSNetWorx partent du principe que l'on utilise la configuration par défaut de la passerelle LUF9.

### 6.1. Raccordement de la passerelle au PC de configuration

Cette étape est obligatoire lors de la mise en œuvre du logiciel permettant de configurer la passerelle, ABC-LUFP Config Tool.

Le raccordement de la passerelle à l'un des ports série (COM) d'un ordinateur nécessite un câble direct PowerSuite et un convertisseur RS232/RS485. Ces deux éléments sont les mêmes que ceux qui permettent de dialoguer avec des variateurs et des démarreurs depuis le logiciel **PowerSuite** et sont tous deux disponibles sur catalogue (réf. : VW3 A8 106).

Veillez à bien utiliser le câble libellé « POWERSUITE » et le convertisseur « RS232 / RS485 PC ». Un câble « ATV28 before 09 / 2001 » et un convertisseur « ATV 58 » sont également fournis avec ces éléments, mais ils ne doivent pas être utilisés dans le cas de la passerelle LUF9.



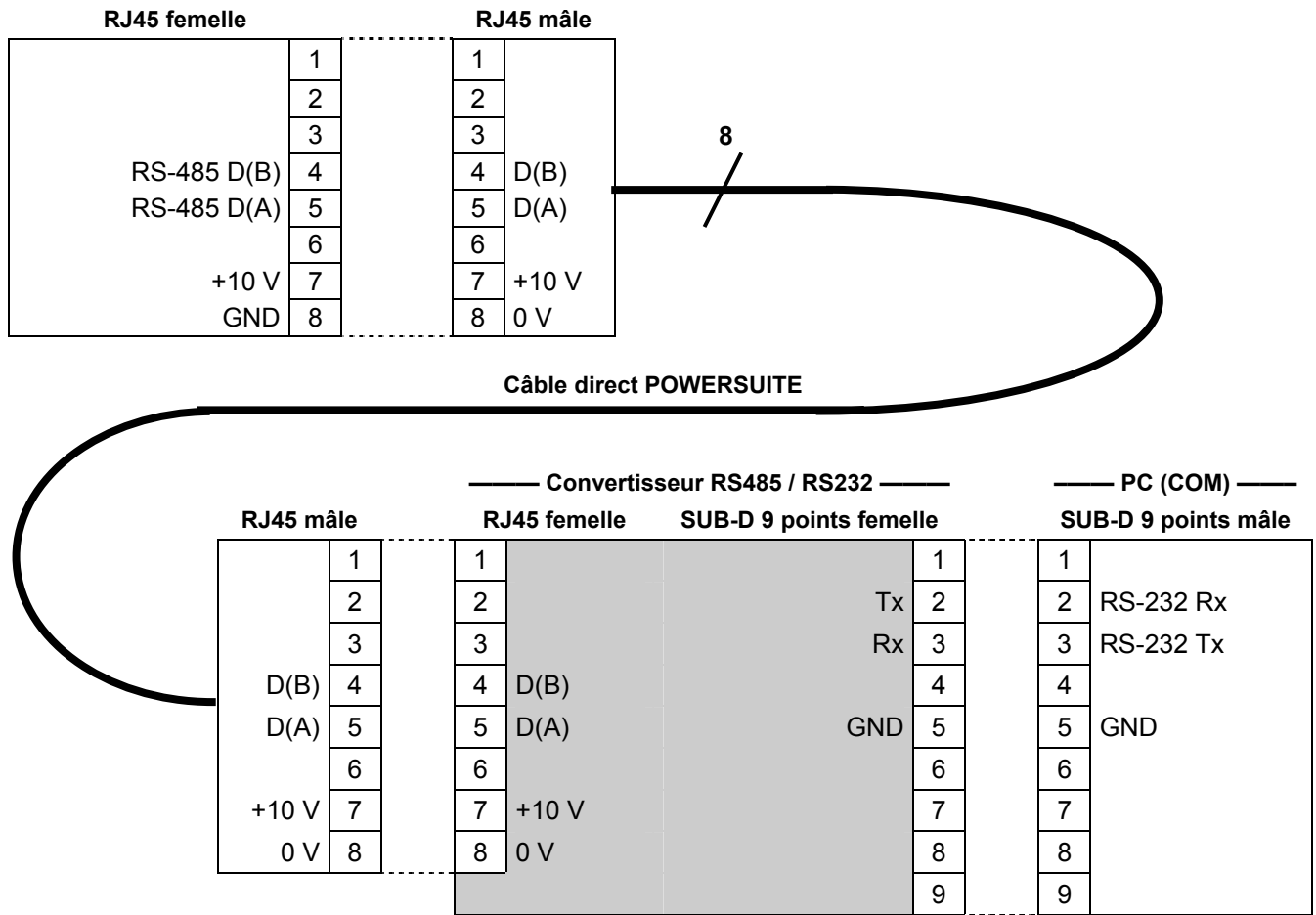
Une fois la passerelle reliée à un PC à l'aide du câble PowerSuite et du convertisseur RS232/RS485, sa configuration peut être modifiée grâce à l'outil de configuration « ABC-LUFP Config Tool ». Ce configurateur permet également d'effectuer quelques diagnostics sur la passerelle.



# 6. Configuration de la passerelle

## 6.1.1. Brochage

— LUF9 (Configuration) —



**REMARQUE** : Le croisement des signaux Rx et Tx entre la passerelle et le PC est représenté au niveau des connecteurs SUB-D 9 points, car au-delà de cette jonction, les signaux RS-232 sont remplacés par les polarisations D(A) et D(B) des signaux RS-485.

## 6.1.2. Protocole de la liaison RS-232

Il n'est pas nécessaire de configurer le port COM du PC, car ABC-LUF9 Config Tool utilise un paramétrage spécifique qui vient remplacer celui du port utilisé. Ce remplacement est temporaire et est annulé dès que ABC-LUF9 Config Tool est fermé.

# 6. Configuration de la passerelle

## 6.2. Installation de ABC-LUFP Config Tool

La configuration minimum requise pour pouvoir utiliser ABC-LUFP Config Tool est la suivante :

- Processeur.....Pentium 133 MHz
- Espace libre sur le disque dur .....10 Mo
- RAM.....08 Mo
- Système d'exploitation.....MS Windows 95 / 98 / ME / NT / 2000 / XP
- Navigateur .....MS Internet Explorer 4.01 SP1

Le programme d'installation de ABC-LUFP Config Tool est disponible sur le site Web <http://www.schneider-electric.com>. Pour l'installer, il suffit de lancer le programme « ABC\_LUFP153.exe » correspondant, puis de suivre les instructions affichées à l'écran.

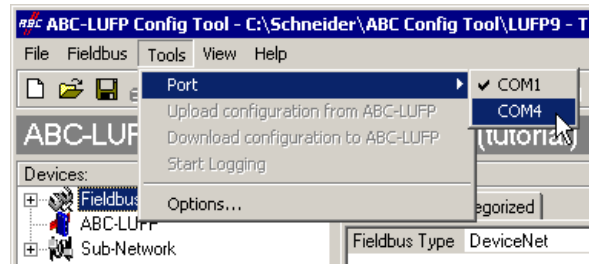
L'utilisation de ABC-LUFP Config Tool est décrite dans un manuel d'utilisation intitulé **AnyBus Communicator – User Manual**. Nous vous recommandons vivement de vous reporter à ce manuel lors de l'utilisation de ABC-LUFP Config Tool, car le présent guide décrit uniquement les différentes possibilités offertes par cet outil dans le cadre de la mise en œuvre de la passerelle LUFP9.

## 6.3. Connexion / déconnexion de la passerelle

Dans ABC-LUFP Config Tool, la connexion à la passerelle doit être effectuée manuellement.


Vous devez d'abord attribuer un port série à ABC-LUFP Config Tool pour établir cette connexion. Dans le menu « Tools », le sous-menu « Port » contient les ports série (COM1, COM2, etc.) actuellement disponibles. Si plusieurs ports COM sont disponibles, sélectionnez celui à utiliser pour le raccordement et la configuration de la passerelle.

Observez l'exemple ci-contre :




**REMARQUE :** Si tous les ports série de votre PC sont déjà utilisés par d'autres applications, vous devez d'abord fermer ABC-LUFP Config Tool, puis libérer un port série en déconnectant, fermant ou arrêtant une de ces applications. Redémarrez ensuite ABC-LUFP Config Tool, car les ports COM sont scrutés uniquement lors du démarrage de ce configurateur. ABC-LUFP Config Tool doit à présent pouvoir utiliser le port COM libéré.

Pour raccorder ABC-LUFP Config Tool à la passerelle :

- cliquez avec le bouton droit de la souris sur « ABC-LUFP », puis cliquez sur « Connect » dans le menu contextuel qui s'affiche, **ou**
- sélectionnez « ABC-LUFP », puis cliquez sur « Connect » dans le menu « ABC-LUFP », **ou**
- cliquez sur le bouton .

Une fois ABC-LUFP Config Tool connecté, vous pouvez le déconnecter de la passerelle en procédant comme suit :





- cliquez avec le bouton droit de la souris sur « ABC-LUFP », puis cliquez sur « Disconnect » dans le menu contextuel qui s'affiche, **ou**
- sélectionnez « ABC-LUFP », puis cliquez sur « Disconnect » dans le menu « ABC-LUFP », **ou**
- cliquez sur le bouton .

La partie la plus à droite de la barre d'état de ABC-LUFP Config Tool indique le mode de connexion actuel du

## 6. Configuration de la passerelle

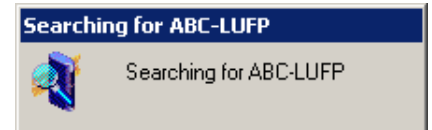
---

configurateur :

Config Line			Mode « connecté » (la DEL de gauche est verte.)
Config Line			Mode « déconnecté » (la DEL de droite est rouge.)

En mode « connecté », ABC-LUFP Config Tool interroge la passerelle régulièrement afin de détecter si la passerelle a été déconnectée.

En cas de déconnexion imprévue, ABC-LUFP Config Tool passe en mode « déconnecté » (la LED devient rouge) et essaie automatiquement de se reconnecter à la passerelle. La fenêtre « Searching for ABC-LUFP » apparaît pendant toute la durée de cette recherche.





En cas d'échec de la recherche, ABC-LUFP Config Tool indique qu'aucun module n'a été trouvé et demande à l'utilisateur s'il souhaite réessayer (« No Module was found, retry? »).

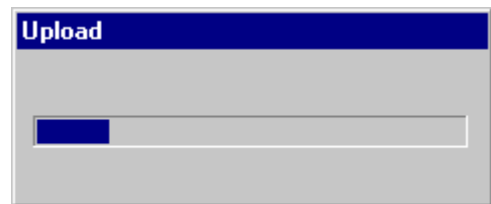
- Si l'utilisateur clique sur le bouton « Cancel », ABC-LUFP Config Tool reste en mode « déconnecté ».
- Si l'utilisateur clique sur le bouton « Retry », ABC-LUFP Config Tool relance la recherche de passerelle.

### 6.4. Importation de la configuration de la passerelle

Avant de pouvoir apporter des modifications à la configuration de la passerelle, vous devez tout d'abord importer sa configuration actuelle. Si cette configuration est déjà présente sur votre disque dur, il vous suffit d'ouvrir le fichier correspondant à cette configuration.

Vérifiez que la passerelle dispose d'une configuration valide et qu'elle fonctionne correctement, c'est-à-dire que la DEL  GATEWAY clignote en vert.

Dans ABC-LUFP Config Tool, sélectionnez « Upload configuration from ABC-LUFP » dans le menu « File » ou cliquez sur le bouton  de la barre d'outils. Une fenêtre intitulée « Upload » s'ouvre alors, et une barre de progression indique l'état d'avancement de la récupération de la configuration de la passerelle. Cette fenêtre disparaît une fois la récupération achevée.



Cette étape est particulièrement importante si vous souhaitez prendre connaissance des détails du contenu de la configuration par défaut de la passerelle, suite à son déballage. Cette configuration pourra ensuite vous servir de modèle pour les modifications que vous apporterez par la suite, vous évitant ainsi d'en créer une de toutes pièces et diminuant les risques d'erreur possibles.


#### REMARQUE :

- Sauvegardez cette configuration sur le disque dur de votre PC afin de pouvoir en disposer à tout moment. Cela vous permettra de reconfigurer la passerelle de manière « propre » dans l'éventualité où sa configuration serait devenue invalide.

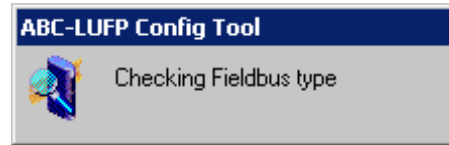
## 6. Configuration de la passerelle

### 6.5. Transfert d'une configuration vers la passerelle

Lorsque vous utilisez ABC-LUFP Config Tool, vous pouvez à tout moment transférer vers la passerelle la configuration qui est en cours d'édition.

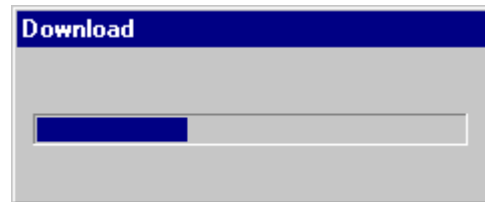
Sélectionnez « Download configuration to ABC-LUFP » dans le menu « File » ou cliquez sur le bouton  de la barre d'outils.

Une phase de vérification du type de la passerelle est initialisée par ABC-LUFP Config Tool.



**REMARQUE :** Pendant cette phase de vérification très rapide, le PC ne doit effectuer aucune autre opération, car cela peut provoquer le blocage apparent de ABC-LUFP Config Tool et le ralentissement du fonctionnement général du PC, et ce pendant plusieurs minutes ! Une fois cette vérification terminée, le PC retrouve sa pleine vitesse et peut être utilisé normalement.

Une fois cette phase achevée, une fenêtre intitulée « Download » s'ouvre et une barre de progression indique l'état d'avancement du transfert de la configuration vers la passerelle.



**REMARQUE :** N'interrompez pas cette opération, car vous seriez obligé de la reprendre depuis le début.

Vérifiez que le transfert s'est correctement déroulé : la DEL  GATEWAY doit clignoter en vert.

Si cette DEL clignote alternativement en rouge et en vert, sauvegardez la configuration que vous étiez en train d'éditer, ouvrez le fichier contenant la configuration par défaut des passerelles LUFP9, puis procédez à son transfert vers la passerelle. Cette procédure permettra de la remettre dans un état initial connu. Vous pourrez ensuite reprendre la configuration précédemment transférée, puis procéder aux corrections nécessaires.

### 6.6. Suivi du contenu de la mémoire de la passerelle

L'une des principales commandes que vous aurez à utiliser lors de la mise en œuvre de la passerelle est la commande qui permet de lire le contenu de la mémoire de la passerelle et de l'afficher dans une fenêtre prévue à cet effet. Elle vous sera particulièrement utile lors de la mise au point de vos configurations et de vos applications automate. Cependant, elle permet de visualiser uniquement les données des champs « Data » et « Preset Data » (mais également celles du champ « Variable Data » réservé aux transactions) configurés dans les éléments « Query » et « Response » d'un seul des esclaves Modbus, plus le contenu des deux registres réservés de la passerelle, situés aux adresses mémoire 0x0000-0x0001 (mot d'état de la passerelle) et 0x0200-0x0201 (mot de commande du maître DeviceNet).

Pour effectuer le suivi du contenu de la mémoire de la passerelle, commencez par sélectionner le nœud qui correspond à l'esclave Modbus dont vous souhaitez visualiser les données, puis exécutez la commande « Monitor » dans le menu dont le nom correspond au nom du nœud préalablement sélectionné. Une fenêtre de suivi apparaît alors. L'exemple de fenêtre qui est reproduit en haut de la page suivante correspond à la visualisation du contenu de la mémoire qui est échangé, dans le cas de la configuration par défaut de la passerelle, avec le départ-moteur « TeSys U n°1 ».

## 6. Configuration de la passerelle

Slave Address	Function code	Starting register address	Number of registers	Checksum
0x01	0x03	0x0000	0x0000	CRC

Slave Address	Function code	Byte count	Data	Checksum
---------------	---------------	------------	------	----------

In Area 32 bytes (512)															
0000	E1	1	0	0											
0008															
0010			1	3	2	0	80								
0018	1	6	2	5F	0	78	7	13							
0020															

Out Area 32 bytes (512)															
0200	60	0	0	0											
0208															
0210			1	3	2	72	0	1							
0218	1	6	2	5F	0	78	7	13							
0220															

General Area 0 bytes (960)															
0400															
0408															
0410															
0418															
0420															

La partie supérieure de cette fenêtre permet de choisir une commande Modbus, d'en éditer le contenu, puis de l'envoyer sur le réseau Modbus (menu « Command »). La réponse sera ensuite affichée dans cette même partie. Reportez-vous au chapitre 2.10 Node monitor du manuel d'utilisation de ABC-LUFP Config Tool, intitulé **AnyBus Communicator – User Manual**, pour obtenir de plus amples informations sur l'utilisation de cette fenêtre.

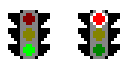
La partie inférieure de cette fenêtre permet de visualiser le contenu de la mémoire de la passerelle, mais uniquement les octets qui sont utilisés dans les trames des requêtes et des réponses des commandes et des transactions configurées pour le nœud sélectionné. Les valeurs des deux mots réservés de la passerelle (adresses 0x0000-0x0001 et 0x0200-0x0201) sont également affichées, quel que soit le nœud sélectionné.

Dans la fenêtre reproduite ci-dessus, les données affichées correspondent aux valeurs situées aux emplacements mémoire désignés par les champs « Data » des commandes et transactions configurées pour le nœud « TeSys U n°1 », c'est-à-dire les commandes suivantes : « Read Holding Registers », « Preset Multiple Registers », « Transactions 1 » et « Transactions 2 ».

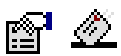
**REMARQUE** : Les données échangées avec l'esclave Modbus précédemment sélectionné sont affichées dans l'ordre octet LSB / octet MSB (lecture de gauche à droite, dans le sens des adresses mémoire croissantes), à condition que l'option « Byte Swap » de l'élément « Data », « Preset Data » ou « Variable Data » de la commande Modbus soit définie sur « Swap 2 bytes » (voir chapitre 6.12.2.5). Dans le cas des deux mots réservés à la gestion du réseau aval Modbus, c'est le contraire : octet MSB / octet LSB.

En revanche, dans le cas du nœud « TeSys U n°1 » uniquement, les données situées à partir des adresses 0x0013, 0x0018, 0x0212 et 0x0218 (voir Annexe B : Contenu de la mémoire DPRAM de la passerelle) suivent le même ordre que le contenu des trames auxquelles elles correspondent (voir Annexe E : Commandes Modbus), du premier au dernier octet (hors checksum), dans le sens des adresses croissantes dans la mémoire de la passerelle. Enfin, les octets 0x001E, 0x001F, 0x021E et 0x021F correspondent aux compteurs de réception et d'émission de ces trames (« Trigger bytes » des transactions 1 et 2). En revanche, tous ces octets sont permutés deux à deux entre la passerelle et le maître DeviceNet.

Les boutons de la barre d'outils de cette fenêtre sont brièvement décrits ci-dessous :



Arrêt / Mise en route des communications avec le nœud sélectionné (voir le menu « Node » ci-après).



Sélection / Envoi de la commande Modbus présentée dans la partie supérieure de la fenêtre (voir le menu « Command » ci-après).



Arrêt / Reprise du rafraîchissement des données affichées dans la partie inférieure de la fenêtre.

## 6. Configuration de la passerelle

---

Les menus de cette fenêtre permettent à l'utilisateur d'effectuer les opérations suivantes :

- Menu « File » - La commande « Exit » permet de fermer la fenêtre « Monitor » et de retourner au ABC-LUFP Config Tool.
  - Menu « Node » :
    - La commande « Start Node » permet de relancer toutes les communications configurées pour le nœud actuellement surveillé. Etant donné qu'un nœud est actif par défaut, cette commande est utile uniquement si l'utilisateur a explicitement arrêté ce nœud avec la commande « Stop Node » (ou avec l'une des commandes décrites au chapitre 5.4, dans la section relative aux champs CC et CD).
    - La commande « Stop Node » permet d'arrêter toutes les communications configurées pour le nœud actuellement surveillé. Par conséquent, toutes les commandes et transactions configurées pour ce nœud sont inhibées. Notez que pour le premier nœud de la configuration par défaut de la passerelle LUF9 (esclave « TeSys U n°1 »), cela entraîne également l'inhibition des deux transactions destinées à lire/écrire n'importe quel paramètre de n'importe quel esclave.
- Remarque :** Les commandes « Stop Node » et « Start Node » peuvent être particulièrement utiles pour isoler un ou plusieurs nœuds afin d'étudier les problèmes de communication Modbus.
- Menu « Command » :
    - La commande « Select Command » permet d'ouvrir la fenêtre « Select Command » dans laquelle l'utilisateur peut sélectionner une commande Modbus (voir chapitre 6.12.2). Une fois la commande sélectionnée, les trames des requêtes et des réponses de cette commande apparaissent dans la partie supérieure de la fenêtre « Monitor ». L'utilisateur peut ensuite modifier la valeur de chaque champ de la trame de requête avant de lancer la commande à l'aide de l'option « Send Command » (voir ci-après).
    - La commande « Send Command » entraîne l'émission de la requête affichée dans la partie supérieure de la fenêtre « Monitor ». Dès que la passerelle reçoit une réponse Modbus, ABC-LUFP Config Tool affiche le contenu correspondant dans la partie supérieure de la fenêtre « Monitor ».
  - Menu « Columns » :
    - L'option « Free » permet de configurer les trois colonnes de surveillance (« In Area », « Out Area » et « General Area ») de façon à ajuster automatiquement leur largeur sur 1 octet (1 octet, 2 octets, 3 octets, etc.) chaque fois que l'utilisateur modifie la largeur de la fenêtre « Monitor ».
    - L'option « 8 Multiple » permet de configurer les trois colonnes de surveillances afin d'ajuster automatiquement leur largeur sur 8 octets (8 ou 16 octets) chaque fois que l'utilisateur modifie la largeur de la fenêtre « Monitor ».
  - Menu « View » :
    - L'option « Hex » permet de configurer les trois colonnes de surveillance afin d'afficher toutes les valeurs surveillées et les adresses mémoire en hexadécimal.
    - L'option « Decimal » permet de configurer les trois colonnes de surveillance afin d'afficher toutes les valeurs surveillées et les adresses mémoire en décimal.

## 6. Configuration de la passerelle

### 6.7. Suppression d'un esclave Modbus

Cette étape permet, par exemple, de libérer un emplacement sur le réseau aval Modbus, appelé « Sub-Network » dans ABC-LUFP Config Tool, afin de remplacer un esclave Modbus par un autre.

En effet, la configuration par défaut de la passerelle lui permet de communiquer avec huit départs-moteurs TeSys U, ce qui constitue le nombre maximum d'esclaves Modbus.

Si la passerelle est utilisée pour gérer les échanges sur un réseau Modbus comportant moins de huit départs-moteurs TeSys U, il est préférable de supprimer de la configuration de la passerelle les départs-moteurs TeSys U redondants. Vous devez effectuer cette opération à l'aide de ABC-LUFP Config Tool.

Si vous utilisez les services aperiodiques de lecture/écriture, n'oubliez pas que ces services sont configurés à l'aide de l'espace mémoire du premier départ-moteur TeSys U configuré. Par conséquent, la suppression du premier départ-moteur TeSys U configuré peut également engendrer la suppression des services aperiodiques de lecture/écriture.

### AVERTISSEMENT

#### PERTE DES COMMUNICATIONS APERIODIQUES

Ne supprimez pas le premier départ-moteur TeSys U configuré si vous utilisez les services aperiodiques de lecture/écriture. La suppression de ce premier élément provoquera également la suppression des services aperiodiques. Puisque ces services permettent de communiquer avec tous les périphériques Modbus configurés, et pas uniquement avec le premier périphérique, vous risquez de perdre les communications avec tous les périphériques et de provoquer un fonctionnement imprévu de l'appareil.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

#### Procédure de suppression d'un esclave Modbus :

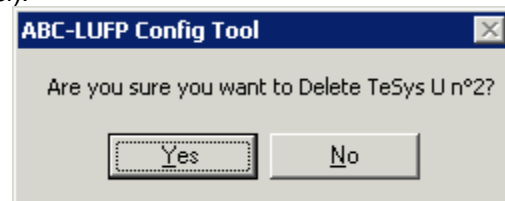
- 1) Sélectionnez le nœud qui correspond à l'esclave Modbus que vous souhaitez supprimer de la configuration. S'il ne reste plus que cet unique nœud dans la configuration, vous ne pourrez pas le supprimer, car le réseau aval Modbus doit comporter au moins un esclave.
- 2) Cliquez, à l'aide du bouton droit de la souris, sur l'icône ou sur le nom de cet esclave Modbus. Un menu apparaît sous le curseur de la souris.

*ou*

Dans le menu principal de ABC-LUFP Config Tool, ouvrez le menu dont le nom correspond au nom du nœud précédemment sélectionné.

- 3) Dans ce menu, cliquez sur la commande « Delete ». La fenêtre de confirmation reproduite ci-dessous apparaît alors, vous demandant de confirmer ou d'annuler la suppression du nœud sélectionné (« TeSys U n°2 » dans le cas de l'exemple présenté ici).

- 4) Si vous confirmez la suppression du nœud, le menu disparaît, ainsi que le nœud auparavant sélectionné. Dans le cas contraire, le nœud sera toujours présent après la disparition de la fenêtre.



**Raccourci clavier :** touche « Suppr ».

#### Ajustement de la mémoire de la passerelle (étape optionnelle) :

Les données auparavant échangées entre la passerelle et l'esclave Modbus qui vient d'être supprimé libéreront des emplacements dans la mémoire de la passerelle. Si vous souhaitez optimiser les échanges entre la mémoire de la passerelle et les entrées/sorties du scanner DeviceNet de l'automate maître, vous devrez modifier la configuration de tous les autres esclaves Modbus afin d'ajuster le contenu de la mémoire de la passerelle.

## 6. Configuration de la passerelle

---

Cependant, ces opérations sont inutiles dans le cas de la suppression d'un unique esclave. A l'inverse, elles deviennent quasiment indispensables lorsque la majeure partie des esclaves Modbus sont supprimés, car ces suppressions morcellent la mémoire de la passerelle.

Reportez-vous au chapitre 6.12, qui décrit l'ensemble des modifications pouvant être apportées à la configuration de chacune des commandes Modbus.

### 6.8. Ajout d'un esclave Modbus

Cette fonctionnalité vous servira à ajouter un esclave Modbus dont le type est différent de celui des autres esclaves Modbus présents dans la configuration. En revanche, si le type de l'esclave correspond à celui de l'un des esclaves précédemment configurés, il est préférable de dupliquer cet esclave plutôt que d'en créer un nouveau.

Une fonctionnalité supplémentaire d'importation/exportation vous permet également de sauvegarder de manière individuelle la configuration complète d'un esclave Modbus, dans le but d'y avoir accès, dans ABC-LUFP Config Tool, depuis n'importe quelle configuration et à n'importe quel moment.

Ces deux fonctionnalités ne sont disponibles qu'à la condition qu'il y ait moins de 8 esclaves Modbus déclarés, ce qui n'est pas le cas de la configuration par défaut, celle-ci comportant 8 départs-moteurs TeSys U.

#### Ajout d'un nouveau type d'esclave Modbus :

Procédez selon l'une des deux méthodes présentées ci-dessous :

- a) Sélectionnez l'élément « Sub-Network », puis exécutez la commande « Add Node » du menu « Sub-Network ». Un nouveau nœud est ajouté à la suite de tous les autres nœuds configurés. Par défaut, son nom est « New Node ».
- b) Sélectionnez l'un des nœuds de l'élément « Sub-Network », puis exécutez la commande « Insert New Node » du menu dont le nom correspond au nom du nœud sélectionné. Un nouveau nœud est ajouté juste avant le nœud sélectionné. Par défaut, son nom est « New Node ».

L'ensemble des étapes permettant de configurer le nouveau nœud sont décrites dans le chapitre 6.11.

#### Duplication d'un esclave Modbus précédemment configuré :

Sélectionnez le nœud qui correspond à l'esclave dont vous comptez recopier la configuration, puis exécutez la commande « Copy » du menu dont le nom correspond au nom du nœud sélectionné. **Raccourci clavier :** « Ctrl C ».

Procédez ensuite selon l'une des deux méthodes présentées ci-dessous :

- a) Sélectionnez l'élément « Sub-Network », puis exécutez la commande « Paste » du menu « Sub-Network ». Un nouveau nœud est ajouté à la suite de tous les autres nœuds configurés. Son nom et l'ensemble de sa configuration sont identiques à ceux du nœud précédemment copié. **Raccourci clavier :** « Ctrl V ».
- b) Sélectionnez l'un des nœuds de l'élément « Sub-Network », puis exécutez la commande « Insert » du menu dont le nom correspond au nœud sélectionné. Un nouveau nœud est ajouté juste avant celui qui est sélectionné. Son nom et l'ensemble de sa configuration sont identiques à ceux du nœud précédemment copié.



## 6. Configuration de la passerelle

---

Le nouveau nœud et le nœud d'origine étant identiques en tous points, vous devrez procéder à la modification (1) du nom du nœud, (2) de l'adresse de l'esclave Modbus correspondant et (3) de l'emplacement des données échangées entre la mémoire de la passerelle et cet esclave Modbus. Reportez-vous aux chapitres 6.11 et 6.12.

### **AVERTISSEMENT**

#### **ADRESSES MODBUS OU PLAGES DE MEMOIRE DE LA PASSERELLE DUPLIQUEES**

Si l'utilisateur choisit d'ajouter un esclave Modbus en recopiant la configuration d'un esclave Modbus existant, il doit modifier l'adresse Modbus du périphérique ajouté et les emplacements mémoire que ce dernier utilise afin d'échanger des données avec la passerelle. Les adresses Modbus ou les emplacements mémoire de la passerelle dupliqués peuvent provoquer des erreurs de communications, l'écriture d'informations incorrectes dans les registres d'un esclave ou l'écriture dans les registres d'un périphérique non souhaité. Chacune de ces erreurs peut provoquer un fonctionnement imprévu de l'appareil.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

#### **Importation/exportation de la configuration d'un esclave Modbus :**

ABC-LUFP Config Tool offre la possibilité de sauvegarder et de charger de manière indépendante la configuration d'un nœud sur le réseau aval « Sub-Network ». Cela vous permettra, par exemple, de constituer une bibliothèque de modèles d'esclaves Modbus, afin de les utiliser dans n'importe quelle configuration.

Pour sauvegarder la configuration d'un esclave Modbus, sélectionnez le nœud auquel il correspond, puis exécutez la commande « Save Node » du menu dont le nom correspond au nom du nœud sélectionné. Une boîte de dialogue vous permettra alors d'en sauvegarder la configuration (export au format XML).

Pour insérer un nœud en prenant pour modèle le fichier XML contenant la configuration d'un esclave Modbus, procédez selon l'une des deux méthodes présentées ci-dessous :

- a) Sélectionnez l'élément « Sub-Network », puis exécutez la commande « Load Node » du menu « Sub-Network ». Une boîte de dialogue vous permet ensuite de choisir un fichier contenant la configuration d'un esclave Modbus (import au format XML). Un nouveau nœud est ajouté à la suite de tous les autres nœuds configurés. Son nom et l'ensemble de sa configuration sont identiques à ceux de l'esclave Modbus, tel qu'il était configuré lors de sa sauvegarde.
- b) Sélectionnez l'un des nœuds de l'élément « Sub-Network », puis exécutez la commande « Insert from File » du menu dont le nom correspond au nom du nœud sélectionné. Un nouveau nœud est ajouté juste avant le nœud sélectionné. Son nom et l'ensemble de sa configuration sont identiques à ceux de l'esclave Modbus, tel qu'il était configuré lors de sa sauvegarde.

Vous devrez ensuite procéder à la modification (1) du nom du nœud, (2) de l'adresse de l'esclave Modbus correspondant et (3) de l'emplacement des données échangées entre la mémoire de la passerelle et cet esclave Modbus. Reportez-vous aux chapitres 6.11 et 6.12.

### **AVERTISSEMENT**

#### **ADRESSES MODBUS OU PLAGES DE MEMOIRE DE LA PASSERELLE DUPLIQUEES**

Si l'utilisateur choisit d'ajouter un esclave Modbus en recopiant la configuration d'un esclave Modbus existant, il doit modifier l'adresse Modbus du périphérique ajouté et les emplacements mémoire que ce dernier utilise afin d'échanger des données avec la passerelle. Les adresses Modbus ou les emplacements mémoire de la passerelle dupliqués peuvent provoquer des erreurs de communications, l'écriture d'informations incorrectes dans les registres d'un esclave ou l'écriture dans les registres d'un périphérique non souhaité. Chacune de ces erreurs peut provoquer un fonctionnement imprévu de l'appareil.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

## 6. Configuration de la passerelle

### 6.9. Modification des données périodiques échangées avec un esclave Modbus

Cette opération consiste à remplacer, à ajouter ou à supprimer des données périodiques échangées avec l'un des esclaves Modbus. Dans le cas de chacune de ces opérations, nous prendrons pour exemple la configuration par défaut de la passerelle LUF9, c'est-à-dire que toute modification précédemment effectuée aura été annulée au début de chaque opération. De plus, les opérations à effectuer sont présentées dans le cadre d'un exemple ciblé.

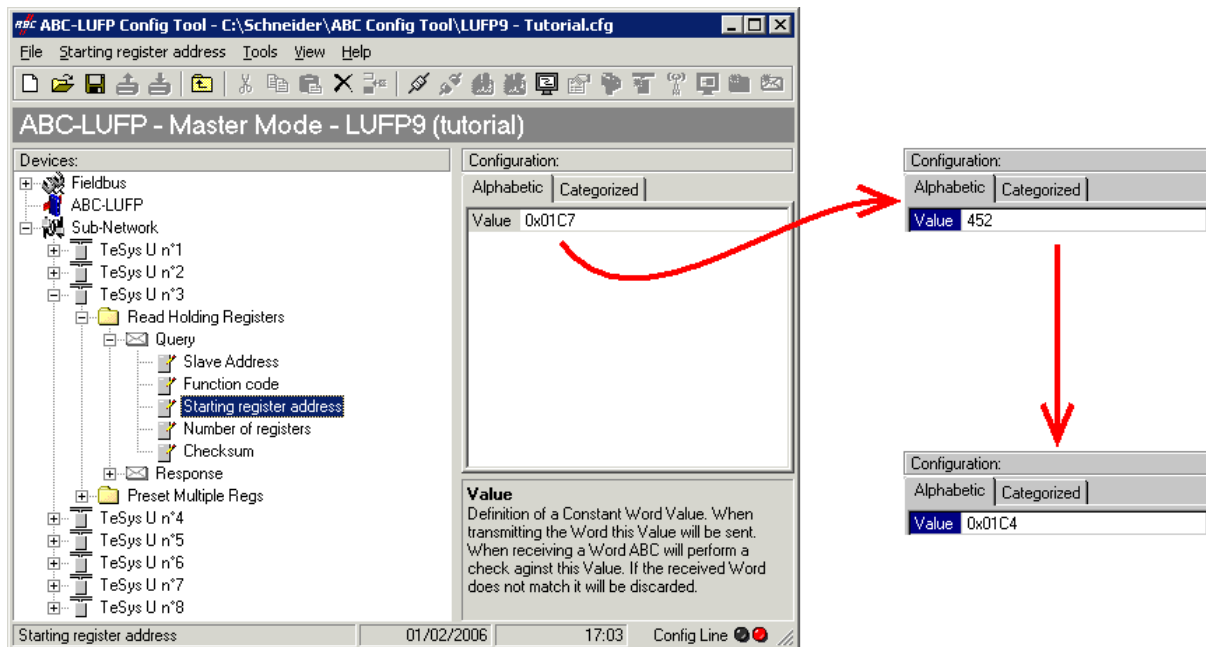
N'oubliez pas de sauvegarder les modifications effectuées ou de transférer l'ensemble de la configuration vers la passerelle. Vous pourrez ainsi vérifier la validité de la configuration, car la passerelle vérifie automatiquement la configuration lorsqu'elle est transmise.

#### 6.9.1. Remplacement d'une donnée périodique d'entrée

Dans notre exemple, nous utiliserons le nœud correspondant au départ-moteur « TeSys U n°3 ». Nous cherchons à remplacer la surveillance du registre « TeSys U Status Register » (adresse 455 = 0x01C7) par la surveillance du registre « 1st Fault Register » (adresse 452 = 0x01C4).

L'opération à effectuer est très simple et consiste uniquement à modifier la valeur de l'élément « Starting register address » de la requête « Query » de la commande « Read Holding Registers » (commande Modbus de lecture des valeurs de plusieurs registres).

Sélectionnez cet élément, puis modifiez sa valeur comme cela est indiqué ci-dessous. Vous pouvez saisir l'adresse du paramètre au format décimal. Elle sera automatiquement convertie au format hexadécimal par ABC-LUFP Config Tool.



Cette opération ne modifie en rien la configuration de la mémoire de la passerelle, car nous n'avons pas besoin de modifier les valeurs des champs « Data length » et « Data location » de l'élément « Data » de la réponse « Response » à la commande précédemment mentionnée. Aucune opération supplémentaire ne sera donc nécessaire, ni dans ABC-LUFP Config Tool, ni sous RSNetWorx.

En revanche, le logiciel de l'automate maître DeviceNet devra tenir compte du changement de la nature de l'entrée correspondante. Dans aucune, Zone mémoire des données d'entrée, la description du mot situé à l'adresse 0x0006 devient « Valeur du 1er registre de défaut du départ-moteur ③ ». Ce mot correspond au mot d'entrée I:1.4 de l'automate (voir chapitre 4.2.6).

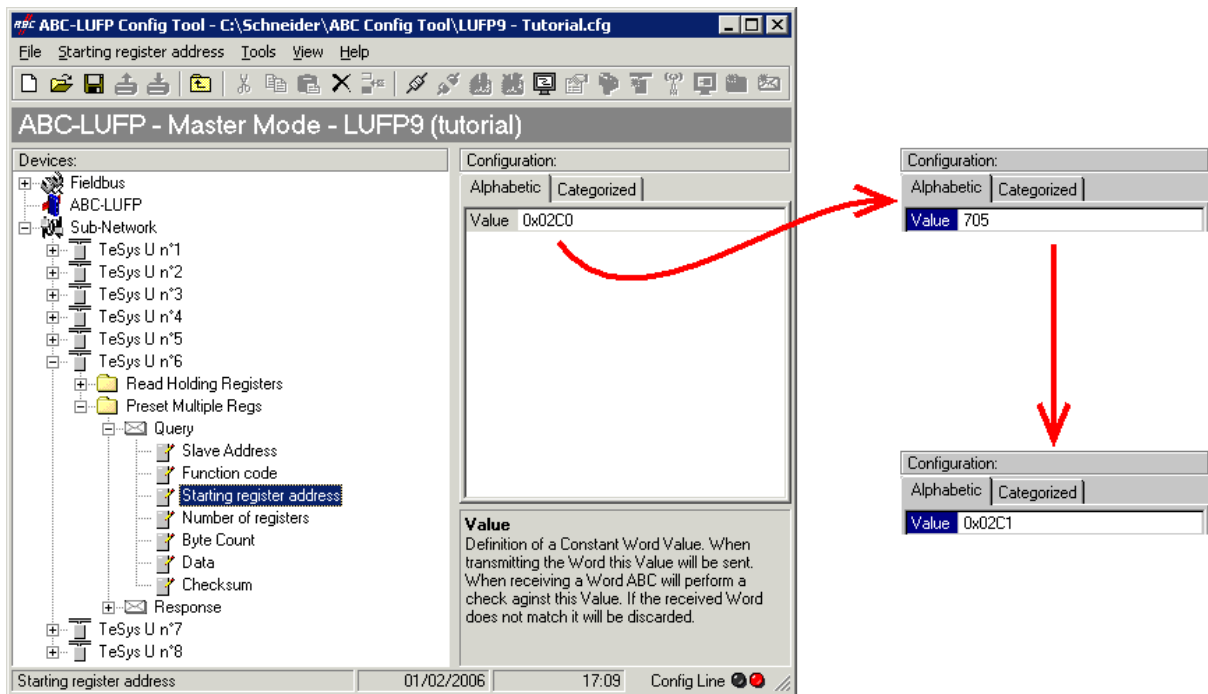
## 6. Configuration de la passerelle

### 6.9.2. Remplacement d'une donnée périodique de sortie

Dans notre exemple, nous utiliserons le nœud correspondant au départ-moteur « TeSys U n°6 ». Nous cherchons à remplacer la commande du registre « Command Register » (adresse 704 = 0x02C0) par la commande du registre « 2nd Command Register » (adresse 705 = 0x02C1).

L'opération consiste à modifier la valeur de l'élément « Starting register address » dans la requête « Query » et dans la réponse « Response » de la commande « Preset Multiple Registers » (commande Modbus d'écriture des valeurs de plusieurs registres).

Sélectionnez l'élément « Starting register address » de la requête « Query », puis modifiez sa valeur comme cela est indiqué ci-dessous. Vous pouvez saisir l'adresse du paramètre au format décimal. Elle sera automatiquement convertie au format hexadécimal par ABC-LUFP Config Tool. **Faites de même pour l'élément « Starting Address » de la réponse « Response »**, car la passerelle vérifie la valeur de ce champ lors de la réception de chaque réponse Modbus. Si la valeur ne correspond pas à celle de la requête, la passerelle ne tiendra pas compte de la réponse.



Cette opération ne modifie en rien le contenu de la mémoire de la passerelle, car nous n'avons pas besoin de modifier les valeurs des champs « Data length » et « Data location » de l'élément « Data » de la requête « Query ». Aucune opération supplémentaire ne sera donc nécessaire, ni dans ABC-LUFP Config Tool, ni sous RSNetWorx.

En revanche, le logiciel de l'automate maître DeviceNet devra tenir compte du changement de la nature de la sortie correspondante. Dans aucune, Zone mémoire des données de sortie, la description du mot situé à l'adresse 0x020C devient « Valeur du 2ème registre de commande du départ-moteur © ». Ce mot correspond au mot de sortie O:1.7 de l'automate (voir chapitre 4.2.7).

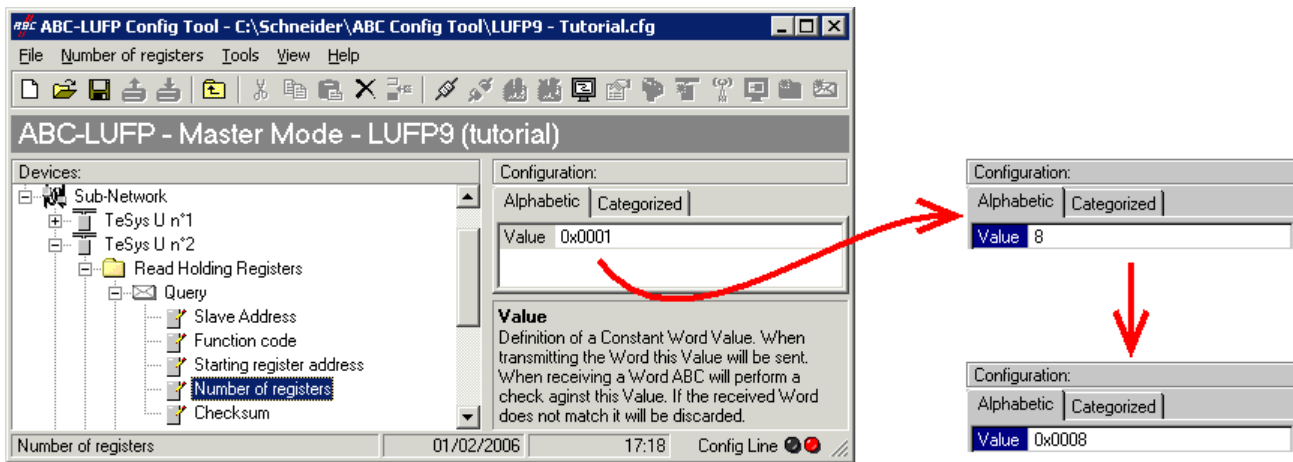
# 6. Configuration de la passerelle

## 6.9.3. Augmentation du nombre des données périodiques d'entrée

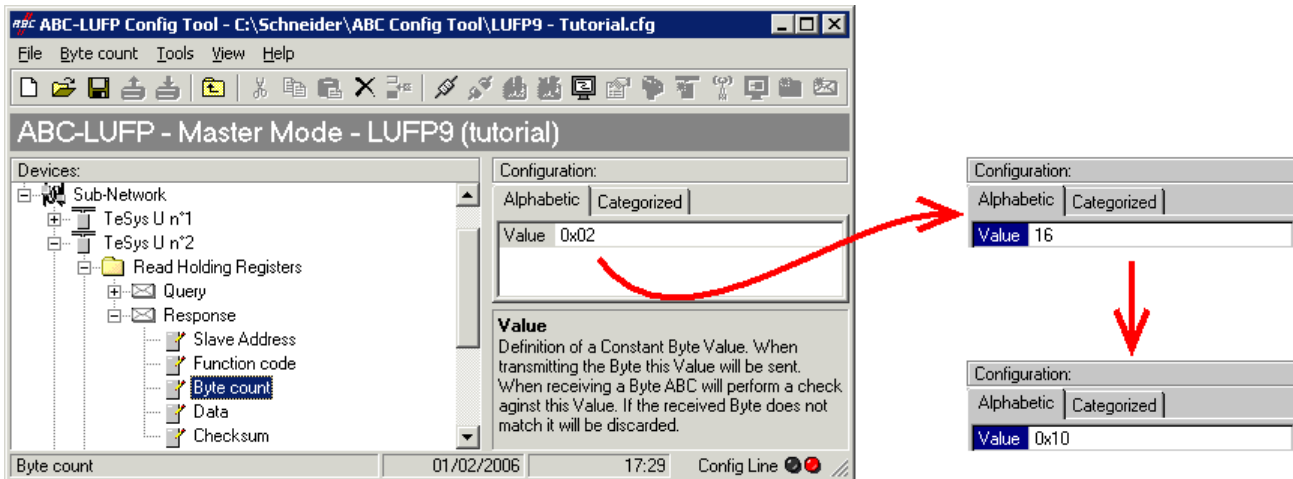
Dans notre exemple, nous utiliserons le nœud correspondant au départ-moteur « TeSys U n°2 ». Nous cherchons à compléter la surveillance de ce départ-moteur en partant du registre actuellement surveillé, c'est-à-dire « TeSys U Status Register » (adresse 455 = 0x01C7), et en allant jusqu'au registre « Reserved »: « 2nd Warning Register » (adresse 462 = 0x01CE). Le nombre de registres surveillés passe donc de 1 à 8.

Dans le cas présent, le nombre d'opérations à effectuer est relativement important. Elles sont décrites, dans l'ordre, ci-après :

- 1) Modification du nombre de registres surveillés : Cette étape consiste à modifier la valeur de l'élément « Number of registers » de la requête « Query » de la commande « Read Holding Registers » (commande Modbus de lecture des valeurs de plusieurs registres). Sélectionnez cet élément, puis modifiez sa valeur comme cela est indiqué ci-dessous. Toute valeur saisie au format décimal sera automatiquement convertie au format hexadécimal par ABC-LUFP Config Tool.



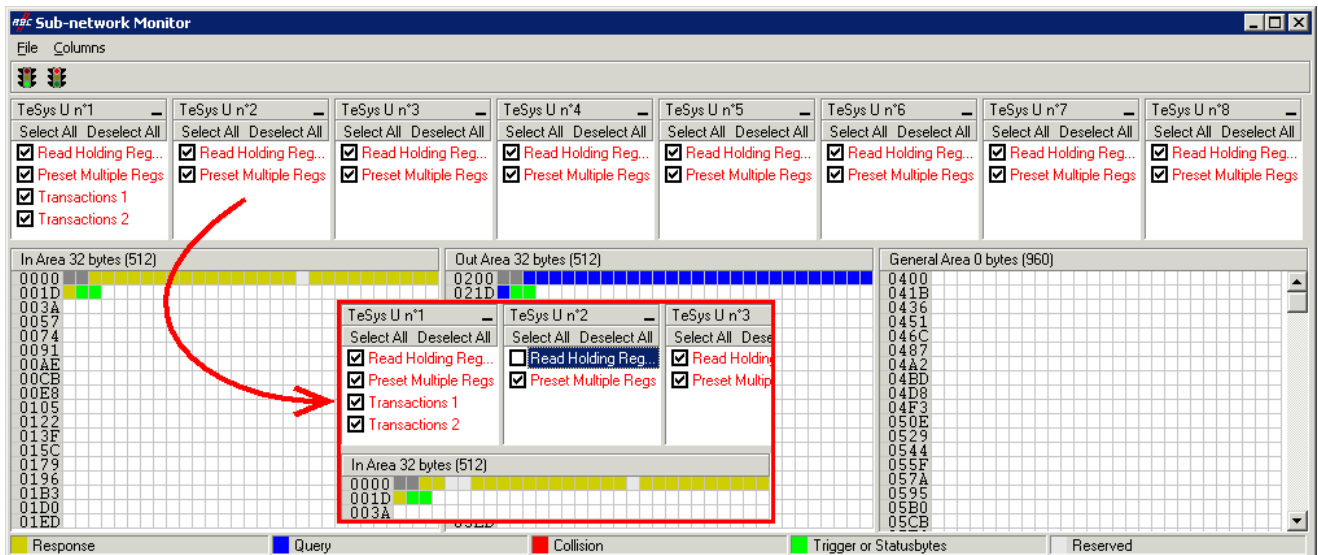
- 2) Modification du nombre d'octets de données dans la réponse Modbus : Le nombre d'octets lus dans la mémoire du départ-moteur « TeSys U n°2 » passe de 2 à 16, puisque le nombre de registres surveillés est passé de 1 à 8. Sélectionnez l'élément « Byte count » de la réponse « Response » et modifiez sa valeur comme cela est indiqué ci-dessous. Toute valeur saisie au format décimal sera automatiquement convertie au format hexadécimal par ABC-LUFP Config Tool.



## 6. Configuration de la passerelle

- 3) Modification de l'emplacement des données Modbus reçues dans la mémoire de la passerelle : Le nombre d'octets récupérés (voir étape précédente) étant passé de 2 à 16, les données Modbus reçues doivent être placées à un endroit différent dans la mémoire de la passerelle, et la taille de la mémoire occupée doit elle aussi être ajustée de manière appropriée.

Si vous n'êtes pas certain de l'occupation mémoire actuelle de la passerelle, sélectionnez l'élément « Sub-Network » et exécutez la commande « Monitor » du menu « Sub-Network ». La fenêtre suivante apparaît alors, vous permettant de consulter l'occupation de la mémoire de la passerelle.



Pour connaître les emplacements mémoire occupés par les données de la commande qui nous intéresse, il suffit de décocher la case qui correspond à la commande « Read Holding Registers » du nœud « TeSys U n°2 », comme cela est indiqué ci-dessus. Nous constatons que les données Modbus reçues en réponse à cette commande occupent 2 octets situés à partir de l'adresse 0x0004.

**REMARQUE** : Les emplacements mémoire 0x0000 et 0x0001 sont réservés (voir chapitre 5). Vous ne pourrez donc pas y placer de données Modbus.

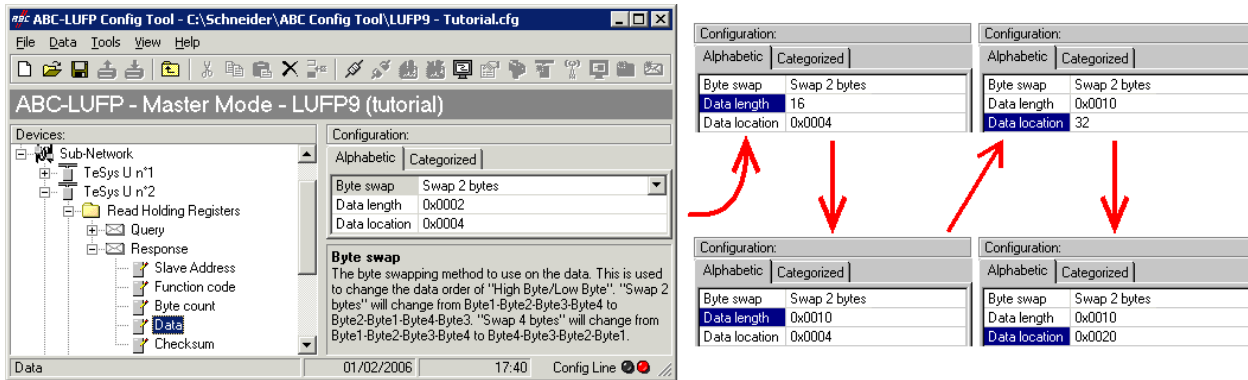
Les tailles indiquées au-dessus des zones graphiques de cette fenêtre (« In Area 32 bytes » et « Out Area 32 bytes ») correspondent aux tailles totales des entrées et des sorties que vous devez observer sous RSNetWorx (voir point 6, page suivante) et configurer pour le scanner DeviceNet (voir point 7)).

Si vous souhaitez placer en mémoire les 16 octets de données Modbus qui seront reçus par la passerelle pour cette commande, une fois les modifications effectuées, il faut soit décaler de 14 octets toutes les autres données reçues, ce qui peut être fastidieux, soit modifier l'emplacement mémoire du bloc des données reçues. Dans le cadre de l'exemple décrit ici, nous utiliserons la seconde solution, bien que la première solution soit préférable, dans le principe, car elle évite de laisser des « trous » dans la mémoire de la passerelle, optimisant ainsi le transfert de l'ensemble des données vers l'automate maître DeviceNet. De plus, le scanner 1747-SDN ne peut échanger que 32 mots d'entrée avec l'automate maître. Le fait de laisser de tels « trous » dans la mémoire de la passerelle est donc déconseillé dans le cas des configurations volumineuses.

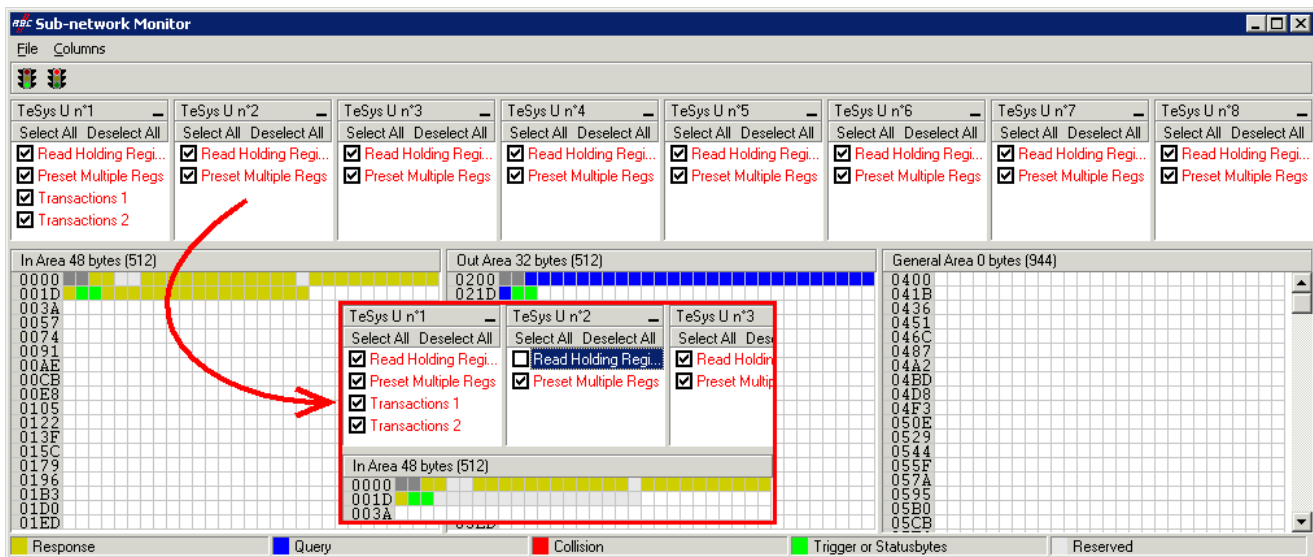
Nous placerons les 16 octets de données à partir de l'adresse 0x0020 (32 au format décimal), c'est-à-dire directement à la suite des données d'entrée de la configuration par défaut de la passerelle.


## 6. Configuration de la passerelle

Fermez la fenêtre « Sub-network Monitor », puis, de retour dans la fenêtre principale de ABC-LUFP Config Tool, sélectionnez l'un après l'autre les champs « Data length » et « Data location » de l'élément « Data » de la réponse « Response » et modifiez leurs valeurs comme indiqué en haut de la page suivante. Toute valeur saisie au format décimal sera automatiquement convertie au format hexadécimal par ABC-LUFP Config Tool.



Afin de vérifier que ces modifications ont été prises en compte dans la configuration, exécutez à nouveau la commande « Monitor » du menu « Sub-Network » :



- 4) Transfert de cette configuration vers la passerelle : Voir chapitre 6.5. Vérifiez que la configuration est valide (clignotement vert de la DEL  GATEWAY).
- 5) Sauvegarde de cette configuration sur le disque dur de votre PC.
- 6) Vérification du paramétrage de la passerelle : Sous RSNetWorx, procédez à la vérification des valeurs des paramètres de la passerelle (voir chapitre 4.2.4). Seule la valeur du paramètre n°7, « Input1 length », doit avoir été modifiée, passant de « 32 bytes » à « 48 bytes ».

**REMARQUE** : Vous devez vérifier si les valeurs des paramètres affichés sont identiques aux tailles des échanges indiquées dans la fenêtre « Sub-network Monitor ». Dans le cas de l'exemple présent, « In Area 48 bytes » implique que la zone « Input1 » commence à l'offset 0 (adresse physique 0x0000) et que sa longueur soit égale à 48 octets. De même, « Out Area 32 bytes » implique que la zone « Output1 » commence à l'offset 0 (adresse physique 0x0200) et que sa longueur soit égale à 32 octets.

## 6. Configuration de la passerelle

- 7) Modification du nombre de données reçues par le scanner DeviceNet : Toujours sous RSNetWorx, modifiez la valeur du nombre de données périodiques reçues par le scanner DeviceNet (voir chapitre 4.2.5). Modifiez la valeur du champ « Rx Size: » de 32 à 48, dans la section « Polled: ».
- 8) Configuration des entrées de l'automate maître DeviceNet : Sous RSNetWorx, établissez une nouvelle correspondance entre les données issues de la passerelle et les entrées automate, selon les besoins de votre application (voir chapitre 4.2.6). Les différentes possibilités offertes par RSNetWorx afin d'établir une correspondance entre les données issues d'un abonné DeviceNet et les entrées automate ne seront pas décrites ici. Reportez-vous à la documentation de ce logiciel afin d'en savoir plus sur cette étape de la mise en œuvre d'un automate maître DeviceNet.

Dans le cadre du présent guide, nous utiliserons la commande « AutoMap » afin d'établir une correspondance « brute » avec toutes les données issues de la passerelle LUF9. Nous obtenons alors la correspondance représentée ci-dessous, dérivée de celle qui est utilisée dans le cas de la configuration par défaut de la passerelle. Les modifications par rapport à la configuration par défaut sont représentées par un fond grisé, tout comme les « emplacements mémoire libres ».

Service	Entrée Automate	Description	
		Bit 0 ..... Bit 7	Bit 8 ..... Bit 15
Gestion du réseau aval Modbus	I:1.1	Mot d'état de la passerelle LUF9 (MSB → 0xxx••)	(LSB → 0x••xx)
Communications périodiques — Surveillance des départs-moteurs TeSys U	I:1.2	Valeur du registre d'état du départ-moteur ①	
	I:1.3	Emplacement mémoire libre	
	I:1.4	Valeur du registre d'état du départ-moteur ③	
	I:1.5	Valeur du registre d'état du départ-moteur ④	
	I:1.6	Valeur du registre d'état du départ-moteur ⑤	
	I:1.7	Valeur du registre d'état du départ-moteur ⑥	
	I:1.8	Valeur du registre d'état du départ-moteur ⑦	
	I:1.9	Valeur du registre d'état du départ-moteur ⑧	
Communications apériodiques — Lecture de la valeur d'un paramètre de départ-moteur (REPNSE)	I:1.10	Emplacement mémoire libre	N° esclave (0x01-0x08)
	I:1.11	Numéro de la fonction (0x03)	Nombre d'octets lus (0x02)
	I:1.12	Valeur du paramètre lu (MSB → 0xxx••) (LSB → 0x••xx)	
Communications apériodiques — Ecriture de la valeur d'un paramètre de départ-moteur (REPNSE)	I:1.13	N° esclave (0x01-0x08)	N° fonction (0x06)
	I:1.14	Adresse du paramètre écrit (MSB → 0xxx••) (LSB → 0x••xx)	
	I:1.15	Valeur du paramètre écrit (MSB → 0xxx••) (LSB → 0x••xx)	
Communications apériodiques (« Trigger bytes » des réponses)	I:1.16	Compteur de réponse de la lecture d'un paramètre	Compteur de réponse de l'écriture d'un paramètre
Communications périodiques — Surveillance du départ-moteur TeSys U ②	I:1.17	Valeur du registre « TeSys U Status Register »	
	I:1.18	Valeur du registre « Complementary Status Register »	
	I:1.19	Valeur du registre « K7 Status Register »	
	I:1.20	Valeur du registre « K7 Status Register 2 (free format) »	
	I:1.21	Valeur du registre « K7 Status Register 3 (free format) »	
	I:1.22	Valeur du registre « Warning Number »	
	I:1.23	Valeur du registre « Warning Register »	
	I:1.24	Valeur du registre « Reserved : 2 <sup>nd</sup> Warning Register »	

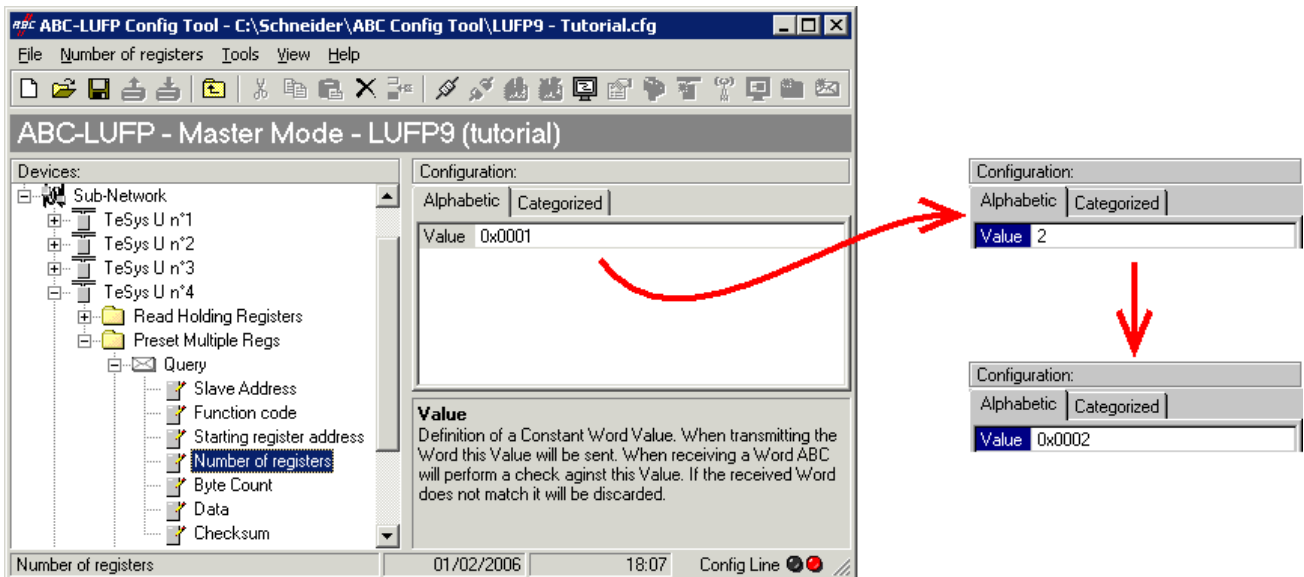
## 6. Configuration de la passerelle

- 9) Transfert de la configuration du scanner DeviceNet : Suite aux modifications apportées à la liste des échanges du scanner DeviceNet, il est nécessaire de la transférer vers le scanner DeviceNet. Reportez-vous au chapitre 4.2.8.

### 6.9.4. Augmentation du nombre des données périodiques de sortie

Dans notre exemple, nous utiliserons le nœud correspondant au départ-moteur « TeSys U n°4 ». Par défaut, nous contrôlons la commande de registre Command Register 704. Pour contrôler également la commande de registre Command Register 705, nous allons effectuer les opérations suivantes.

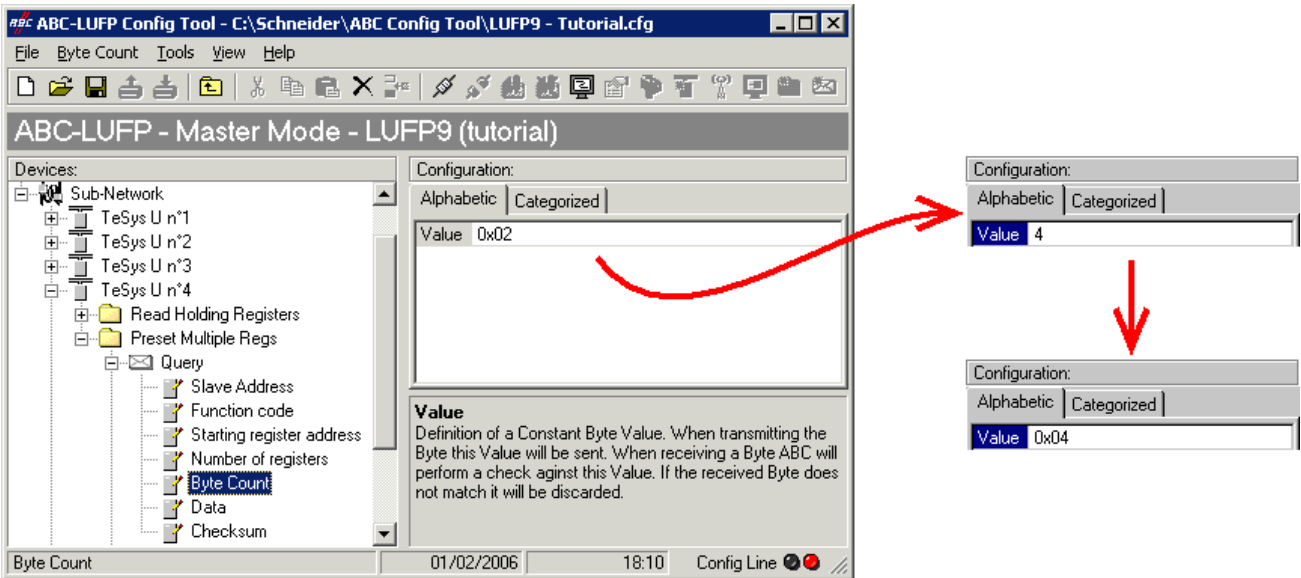
- 1) Modification du nombre de registres commandés : Cette étape consiste à modifier la valeur de l'élément « Number of registers » dans la requête « Query » et dans la réponse « Response » de la commande « Preset Multiple Registers » (commande Modbus d'écriture des valeurs de plusieurs registres). Commencez par sélectionner l'élément « N° of Registers » de la requête « Query », puis modifiez sa valeur comme indiqué ci-dessous. Toute valeur saisie au format décimal sera automatiquement convertie au format hexadécimal par ABC-LUFP Config Tool. Faites de même pour l'élément « N° of Registers » de la réponse « Response », car la passerelle vérifie la valeur de ce champ lors de la réception de chaque réponse Modbus. Si la valeur ne correspond pas à celle de la requête, la passerelle ne tiendra pas compte de la réponse.





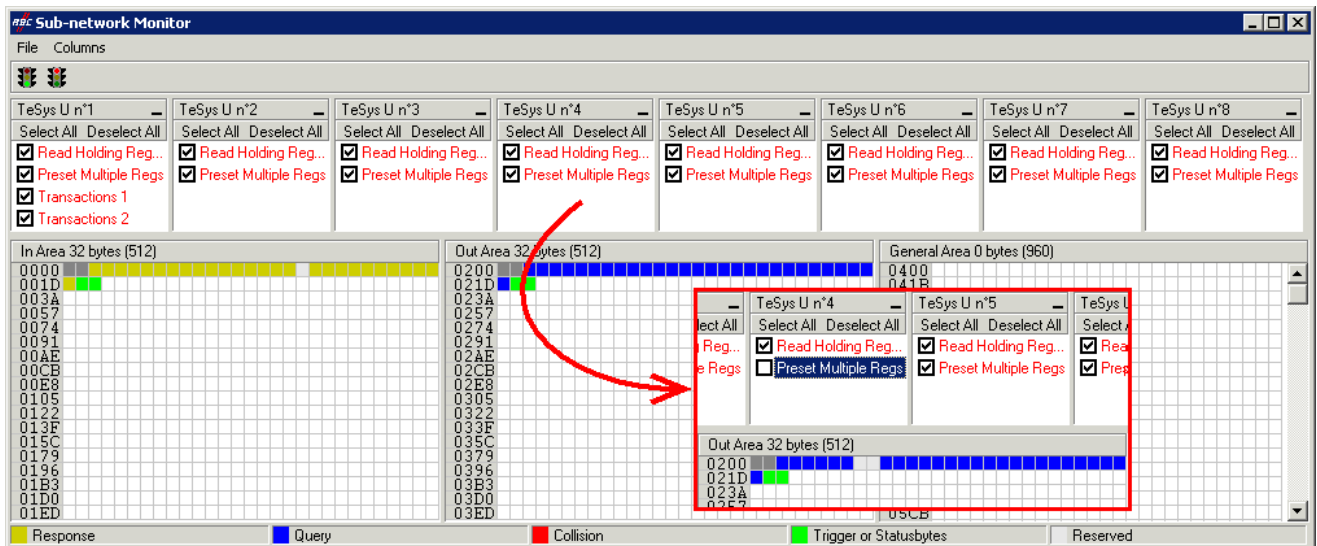
## 6. Configuration de la passerelle

- 2) Modification du nombre d'octets de données dans la requête Modbus : Le nombre d'octets écrits dans la mémoire du départ-moteur « TeSys U n°4 » passe de 2 à 4, puisque le nombre de registres commandés est passé de 1 à 2. Sélectionnez l'élément « Byte count » de la requête « Query » et modifiez sa valeur comme cela est indiqué ci-dessous. Toute valeur saisie au format décimal sera automatiquement convertie au format hexadécimal par ABC-LUFP Config Tool.



- 3) Modification de l'emplacement des données Modbus transmises dans la mémoire de la passerelle : Le nombre d'octets écrits (voir étape précédente) étant passé de 2 à 4, les données Modbus à transmettre au départ-moteur « TeSys U n°4 » doivent être placées à un endroit différent dans la mémoire de la passerelle, et la taille de la mémoire occupée doit elle aussi être ajustée de manière appropriée.

Si vous n'êtes pas certain de l'occupation mémoire actuelle de la passerelle, sélectionnez l'élément « Sub-Network » et exécutez la commande « Monitor » du menu « Sub-Network ». La fenêtre ci-dessous apparaît alors, vous permettant de consulter l'occupation de la mémoire de la passerelle.



## 6. Configuration de la passerelle

Pour connaître les emplacements mémoire occupés par les données de la commande qui nous intéresse, il suffit de décocher la case qui correspond à la commande « Preset Multiple Registers » du nœud « TeSys U n°4 », comme cela est indiqué ci-dessus. Nous constatons que les données Modbus transmises avec la requête correspondant à cette commande occupent 2 octets situés à partir de l'adresse 0x0208.

**REMARQUE** : Les emplacements mémoire 0x0200 et 0x0201 sont réservés (voir chapitre 5). Vous ne pourrez donc pas y placer de données Modbus.

Les tailles indiquées au-dessus des zones graphiques de cette fenêtre (« In Area 32 bytes » et « Out Area 32 bytes ») correspondent aux tailles totales des entrées et des sorties que vous devez observer sous RSNetWorx (voir point 6, page suivante) et configurer pour le scanner DeviceNet (voir point 7)).

Si vous souhaitez placer en mémoire les 4 octets de données Modbus qui seront transmis par la passerelle pour cette commande, une fois les modifications effectuées, il faut soit décaler de 2 octets toutes les autres données transmises, ce qui peut être fastidieux, soit modifier l'emplacement mémoire du bloc des données transmises. Dans le cadre de l'exemple décrit ici, nous utiliserons la seconde solution, bien que la première solution soit préférable, dans le principe, car elle évite de laisser des « trous » dans la mémoire de la passerelle, optimisant ainsi le transfert de l'ensemble des données depuis l'automate maître DeviceNet. De plus, le scanner 1747-SDN ne peut échanger que 32 mots de sortie avec l'automate maître. Le fait de laisser de tels « trous » dans la mémoire de la passerelle est donc déconseillé dans le cas des configurations volumineuses.

Lorsque vous sélectionnez une valeur pour le champ « Data Location », les données doivent être situées à des adresses paires afin d'aligner les données Modbus (au format 16 bits) sur les sorties O:1.x du scanner DeviceNet. Si les données ne sont pas situées à des adresses paires, les valeurs prévues pour les registres Modbus peuvent être réparties sur deux mots de l'automate DeviceNet. Ceci complique considérablement la programmation de l'application, car celle-ci peut être contrainte d'analyser un mot de l'automate pour l'octet Modbus LSB et un autre pour l'octet Modbus MSB. Si ce problème n'est pas géré correctement, il est possible de lire et d'écrire des valeurs de données erronées sur les esclaves Modbus.

### **AVERTISSEMENT**

#### **RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT**

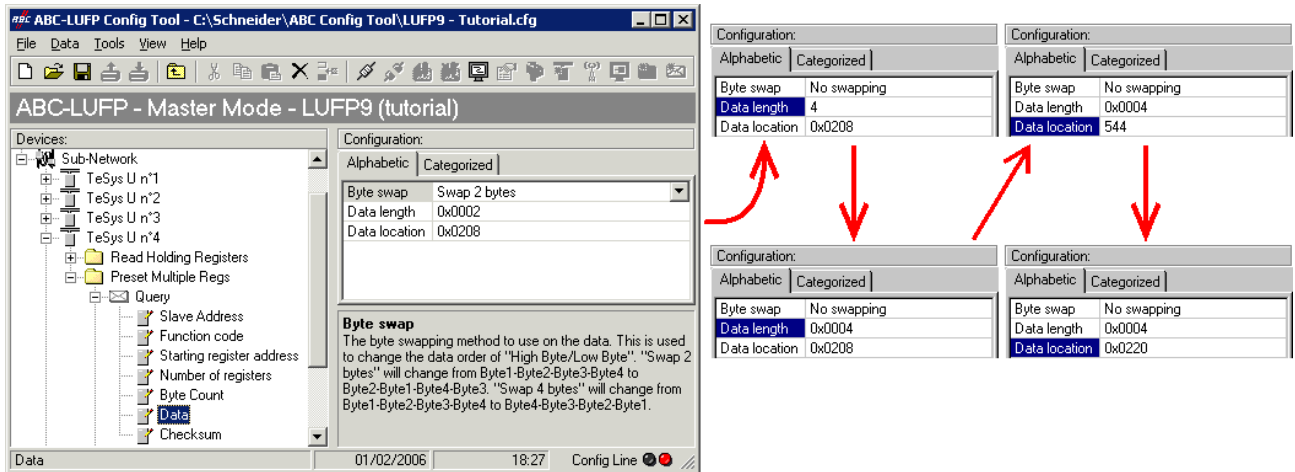
L'utilisateur doit utiliser des valeurs paires pour le champ « Data Location ». La sélection de valeurs de données impaires complique la programmation de l'application et augmente les risques d'écriture ou de lecture de valeurs Modbus incorrectes sur ou depuis les périphériques esclaves. Selon la configuration de l'utilisateur, cela peut provoquer un fonctionnement imprévu de l'appareil.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

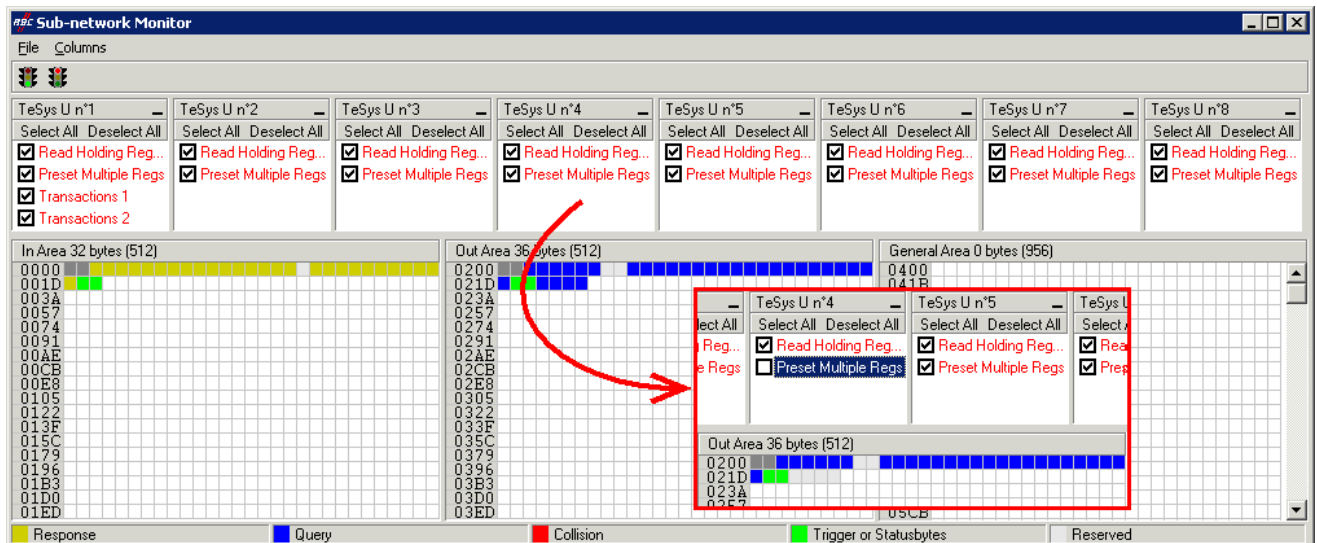
Nous placerons les 4 octets de données à partir de l'adresse 0x0220 (544 au format décimal).

## 6. Configuration de la passerelle

Fermez la fenêtre « Sub-network Monitor », puis, de retour dans la fenêtre principale de ABC-LUFP Config Tool, sélectionnez l'un après l'autre les champs « Data length » et « Data location » de l'élément « Data » de la requête « Query » et modifiez leurs valeurs comme indiqué en haut de la page suivante. Toute valeur saisie au format décimal sera automatiquement convertie au format hexadécimal par ABC-LUFP Config Tool.



Afin de vérifier que ces modifications ont été prises en compte dans la configuration, exécutez à nouveau la commande « Monitor » du menu « Sub-Network » :



- 4) Transfert de cette configuration vers la passerelle : Voir chapitre 6.5. Vérifiez que la configuration est valide (clignotement vert de la DEL GATEWAY).
- 5) Sauvegarde de cette configuration sur le disque dur de votre PC.
- 6) Vérification du paramétrage de la passerelle : Sous RSNetWorx, procédez à la vérification des valeurs des paramètres de la passerelle (voir chapitre 4.2.4). Seule la valeur du paramètre n°19, « Output1 length », doit avoir été modifiée, passant de « 32 bytes » à « 36 bytes ».

**REMARQUE** : Vous devez vérifier si les valeurs des paramètres affichés sont identiques aux tailles des échanges indiquées dans la fenêtre « Sub-network Monitor ». Dans le cas de l'exemple présent, « In Area 32 bytes » implique que la zone « Input1 » commence à l'offset 0 (adresse physique 0x0000) et que sa longueur soit égale à 32 octets. De même, « Out Area 36 bytes » implique que la zone « Output1 » commence à l'offset 0 (adresse physique 0x0200) et que sa longueur soit égale à 36 octets.

## 6. Configuration de la passerelle

- 7) Modification du nombre de données émises par le scanner DeviceNet : Toujours sous RSNetWorx, modifiez la valeur du nombre de données périodiques transmises par le scanner DeviceNet (voir chapitre 4.2.5). Modifiez la valeur du champ « Tx Size: » de 32 à 36, dans la section « Polled: ».
- 8) Configuration des sorties de l'automate maître DeviceNet : Sous RSNetWorx, établissez une nouvelle correspondance entre les données transmises à la passerelle et les sorties automate, selon les besoins de votre application (voir chapitre 4.2.7). Les différentes possibilités offertes par RSNetWorx afin d'établir une correspondance entre les données transmises à un abonné DeviceNet et les sorties automate ne seront pas décrites ici. Reportez-vous à la documentation de ce logiciel afin d'en savoir plus sur cette étape de la mise en œuvre d'un automate maître DeviceNet.

Dans ce guide, nous utiliserons la commande « AutoMap » afin d'établir une correspondance « brute » avec toutes les données transmises à la passerelle LUF9. Nous obtenons alors la correspondance représentée ci-dessous, dérivée de celle qui est utilisée dans le cas de la configuration par défaut de la passerelle. Les modifications par rapport à la configuration par défaut sont représentées par un fond grisé, tout comme les « emplacements mémoire libres ».

Service	Sortie Automate	Description	
		Bit 0..... Bit 7	Bit 8.....Bit 15
Gestion du réseau aval Modbus	O:1.1	Mot de commande du maître DeviceNet (MSB → 0xxx••)	(LSB → 0x••xx)
Communications périodiques — Commande des départs-moteurs TeSys U	O:1.2	Valeur du registre de commande du départ-moteur ①	
	O:1.3	Valeur du registre de commande du départ-moteur ②	
	O:1.4	Valeur du registre de commande du départ-moteur ③	
	O:1.5	Emplacement mémoire libre	
	O:1.6	Valeur du registre de commande du départ-moteur ⑤	
	O:1.7	Valeur du registre de commande du départ-moteur ⑥	
	O:1.8	Valeur du registre de commande du départ-moteur ⑦	
	O:1.9	Valeur du registre de commande du départ-moteur ⑧	
Communications aperiodiques — Lecture de la valeur d'un paramètre de départ-moteur (REQUETE)	O:1.10	N° esclave (0x01-0x08)	N° fonction (0x03)
	O:1.11	Adresse du paramètre à lire (MSB → 0xxx••)	(LSB → 0x••xx)
	O:1.12	Nombre de paramètres à lire (MSB → 0x00••)	(LSB → 0x••01)
Communications aperiodiques — Ecriture de la valeur d'un paramètre de départ-moteur (REQUETE)	O:1.13	N° esclave (0x01-0x08)	N° fonction (0x06)
	O:1.14	Adresse du paramètre à écrire (MSB → 0xxx••)	(LSB → 0x••xx)
	O:1.15	Valeur du paramètre à écrire (MSB → 0xxx••)	(LSB → 0x••xx)
Communications aperiodiques (« Trigger bytes » des requêtes)	O:1.16	Compteur de requête de la lecture d'un paramètre	Compteur de requête de l'écriture d'un paramètre
Communications périodiques — Surveillance du départ-moteur TeSys U ④	O:1.17	Valeur du registre de commande : « Command Register »	
	O:1.18	Valeur du deuxième registre de commande : « 2nd Command Register »	

- 9) Transfert de la configuration du scanner DeviceNet : Suite aux modifications apportées à la liste des échanges du scanner DeviceNet, il est nécessaire de la transférer vers le scanner DeviceNet. Reportez-vous au chapitre 4.2.8.

## 6. Configuration de la passerelle

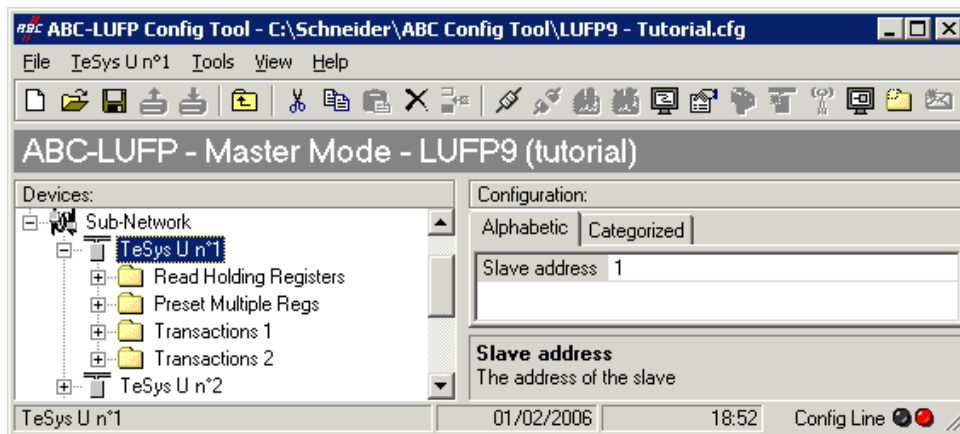
### 6.10. Suppression des données apériodiques de paramétrage

Si votre application automate n'a pas besoin du service apériodique de paramétrage des esclaves Modbus, vous pouvez supprimer les commandes qui lui sont associées. Si vous comptez également ajouter des données Modbus, et donc utiliser de nouveaux emplacements dans la mémoire de la passerelle, il est préférable que vous supprimiez les commandes de paramétrage dès le début afin d'en réutiliser les emplacements mémoire.

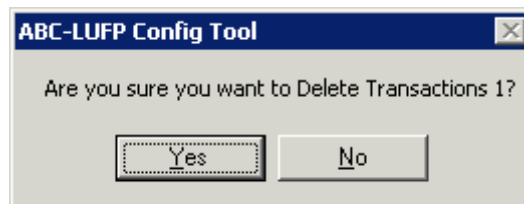
En revanche, si la seule opération de configuration que vous souhaitez effectuer sur la passerelle LUFF9 consiste à ne pas utiliser le service apériodique de paramétrage, vous pouvez vous contenter de ne pas utiliser ce service sous RSNetWorx. Passez directement à l'étape 8.

Si vous décidez de supprimer les commandes apériodiques, vous devez effectuer les opérations suivantes :

- 1) Affichage des commandes de paramétrage : Sélectionnez le tout premier nœud du réseau aval Modbus, « TeSys U n°1 », et développez l'arborescence de ses commandes et de ses transactions. L'affichage obtenu est reproduit ci-dessous :



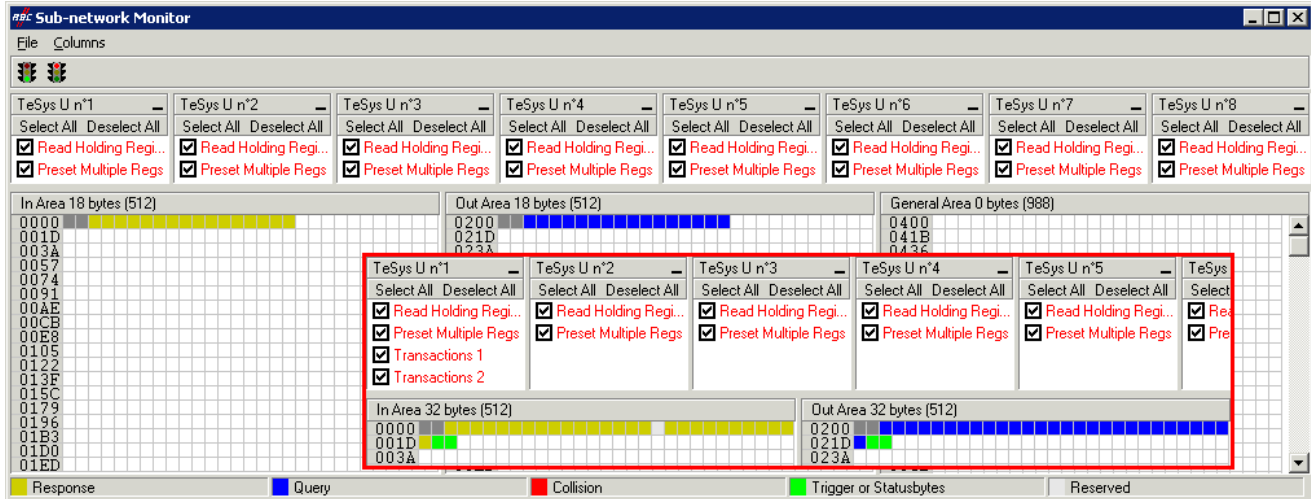
- 2) Suppression de la commande de lecture d'un paramètre : Sélectionnez la commande personnalisée « Transactions 1 » et supprimez-la à l'aide de la touche « Suppr » (ou la commande « Delete » du menu dont le nom correspond au nom du nœud sélectionné). Une demande de confirmation apparaît alors, vous proposant de procéder ou non à la suppression de la commande « Transactions 1 ». Dans le cas présent, validez la suppression à l'aide du bouton « Yes ».



- 3) Suppression de la commande d'écriture d'un paramètre : De retour dans la fenêtre principale de ABC-LUFFP Config Tool, la commande « Transactions 1 » a été supprimée. La seconde commande personnalisée, « Transactions 2 », est automatiquement renommée en « Transactions 1 », mais conserve l'ensemble de son paramétrage. Supprimez-la à son tour, de la même manière que pour la commande précédente. Cette opération n'a aucune conséquence sur les autres nœuds.


## 6. Configuration de la passerelle

- 4) Vérification de la nouvelle occupation mémoire : Si vous souhaitez vérifier la nouvelle occupation mémoire de la passerelle, sélectionnez l'élément « Sub-Network » et exécutez la commande « Monitor » du menu « Sub-Network ». La fenêtre suivante apparaît alors, vous permettant de consulter la nouvelle occupation de la mémoire de la passerelle par les données Modbus. La partie encadrée en rouge représente l'occupation de la mémoire avant la suppression des deux commandes de paramétrage. Elle a été incrustée dans l'illustration présentée ci-dessous pour vous permettre de constater les effets des suppressions effectuées.



Vous constaterez que la section « TeSys U n°1 » ne comporte plus que les deux commandes Modbus communes aux huit départs-moteurs TeSys U, et que les emplacements mémoire qui correspondaient au deux commandes personnalisées sont désormais libres.

**REMARQUE** : L'emplacement mémoire libre situé à l'adresse 0x0012 de la mémoire de la passerelle ne fait désormais plus partie des entrées de la passerelle, car il n'y a plus aucune donnée d'entrée utilisée au-delà de cette adresse.

- 5) Transfert de cette configuration vers la passerelle : Voir chapitre 6.5. Vérifiez que la configuration est valide (clignotement vert de la DEL  GATEWAY).
- 6) Sauvegarde de cette configuration sur le disque dur de votre PC.
- 7) Vérification du paramétrage de la passerelle : Sous RSNetWorx, procédez à la vérification des valeurs des paramètres de la passerelle (voir chapitre 4.2.4). La valeur du paramètre n°7, « Input1 length », doit avoir été modifiée, passant de « 32 bytes » à « 18 bytes ». La valeur du paramètre n°19, « Output1 length », passe de « 32 bytes » à « 18 bytes ».
- 8) Modification du nombre de données reçues et du nombre de données émises par le scanner DeviceNet : Toujours sous RSNetWorx, modifiez le nombre de données périodiques reçues et le nombre de données périodiques transmises par le scanner DeviceNet (voir chapitre 4.2.5). Dans la section « Polled: », modifiez la valeur du champ « Rx Size: » de 32 à 18 et la valeur du champ « Tx Size: » de 32 à 18.
- 9) Configuration des entrées et des sorties de l'automate maître DeviceNet : Sous RSNetWorx, établissez une nouvelle correspondance entre les données issues de la passerelle et les entrées automate (voir chapitre 4.2.6). Effectuez la même opération pour la correspondance entre les données transmises à la passerelle et les sorties automate (voir chapitre 4.2.7).

Nous obtenons alors les deux correspondances représentées sur la page suivante, dérivées de celles qui sont utilisées dans le cas de la configuration par défaut de la passerelle.

## 6. Configuration de la passerelle

Service	Entrée Automate	Description	
		Bit 0 ..... Bit 7	Bit 8 ..... Bit 15
Gestion du réseau aval Modbus	I:1.1	Mot d'état de la passerelle LUF9 (MSB → 0xxx••)   (LSB → 0x••xx)	
Communications périodiques — Surveillance des départs-moteurs TeSys U	I:1.2	Valeur du registre d'état du départ-moteur ①	
	I:1.3	Valeur du registre d'état du départ-moteur ②	
	I:1.4	Valeur du registre d'état du départ-moteur ③	
	I:1.5	Valeur du registre d'état du départ-moteur ④	
	I:1.6	Valeur du registre d'état du départ-moteur ⑤	
	I:1.7	Valeur du registre d'état du départ-moteur ⑥	
	I:1.8	Valeur du registre d'état du départ-moteur ⑦	
	I:1.9	Valeur du registre d'état du départ-moteur ⑧	

Service	Sortie Automate	Description	
		Bit 0 ..... Bit 7	Bit 8 ..... Bit 15
Gestion du réseau aval Modbus	O:1.1	Mot de commande du maître DeviceNet (MSB → 0xxx••)   (LSB → 0x••xx)	
Communications périodiques — Commande des départs-moteurs TeSys U	O:1.2	Valeur du registre de commande du départ-moteur ①	
	O:1.3	Valeur du registre de commande du départ-moteur ②	
	O:1.4	Valeur du registre de commande du départ-moteur ③	
	O:1.5	Valeur du registre de commande du départ-moteur ④	
	O:1.6	Valeur du registre de commande du départ-moteur ⑤	
	O:1.7	Valeur du registre de commande du départ-moteur ⑥	
	O:1.8	Valeur du registre de commande du départ-moteur ⑦	
	O:1.9	Valeur du registre de commande du départ-moteur ⑧	

10) Transfert de la configuration du scanner DeviceNet: Suite aux modifications apportées à la liste des échanges du scanner DeviceNet, il est nécessaire de la transférer vers le scanner DeviceNet. Reportez-vous au chapitre 4.2.8.

### 6.11. Modification de la configuration d'un esclave Modbus

La configuration d'un esclave Modbus lui-même reste très simple car elle concerne uniquement le nom et l'adresse Modbus du nœud auquel il correspond. En revanche, la configuration des commandes Modbus est beaucoup plus compliquée et fait l'objet d'un chapitre à part entière (voir chapitre 6.12).

La modification de la configuration d'un esclave Modbus est nécessaire lorsque vous ajoutez un nouvel équipement Modbus (voir chapitre 6.8).

La modification du nom du nœud correspondant à un esclave Modbus permet de le distinguer des autres nœuds lorsque la configuration de ses commandes Modbus a été modifiée.

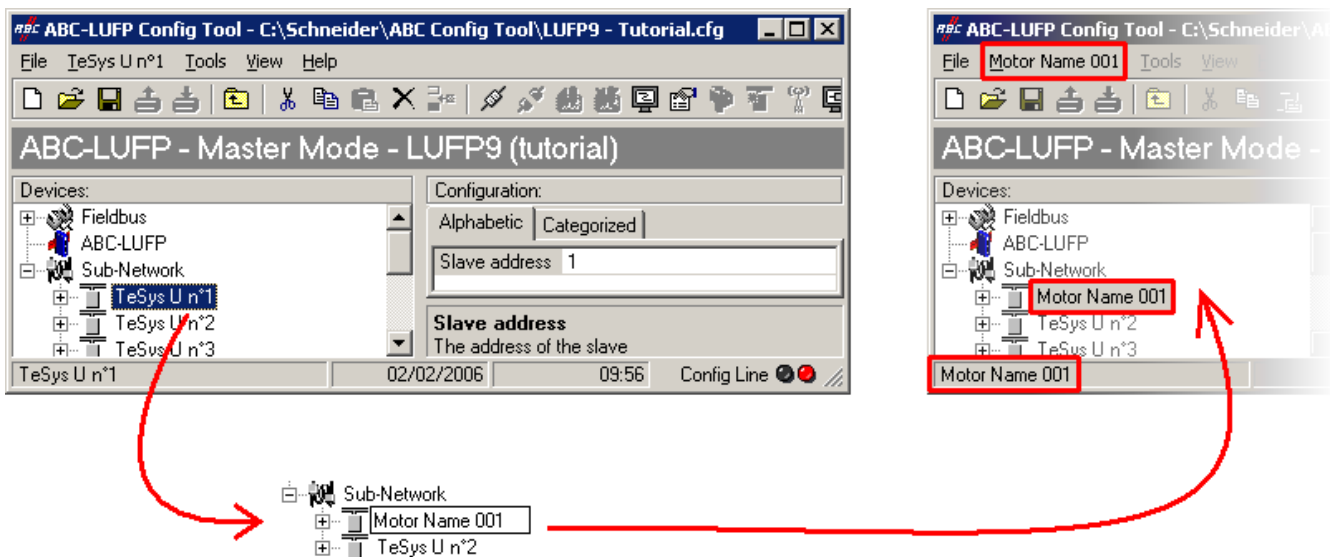
## 6. Configuration de la passerelle

### 6.11.1. Modification du nom d'un esclave Modbus

Pour effectuer cette opération, il suffit de sélectionner le nœud qui correspond à l'esclave Modbus concerné (section « Devices: ») et d'effectuer l'une des quatre opérations suivantes :

- cliquez avec le bouton droit de la souris sur le nœud, puis cliquez sur « Rename » dans le menu contextuel qui s'affiche, **ou**
- sélectionnez le nœud et cliquez sur son nom, **ou**
- sélectionnez le nœud, puis cliquez sur « Rename » dans le menu dont le nom correspond au nom du nœud, **ou**
- utilisez la touche de fonction F2.

Après validation du nouveau nom (touche « Entrée » ou clic en dehors du nom de nœud), il sera mis à jour dans la barre de menus et la barre d'état de ABC-LUFP Config Tool. Un exemple est donné ci-après. Les trois cadres rouges représentés sur cet exemple indiquent les conséquences de la modification apportée.



### 6.11.2. Modification de l'adresse d'un esclave Modbus

Pour effectuer cette opération, il suffit de sélectionner le nœud qui correspond à l'esclave Modbus concerné (section « Devices: »), de cliquer sur la valeur de l'adresse actuelle (valeur du champ « Slave address », dans la section « Configuration: »), puis de la modifier.

**REMARQUE :** L'adresse d'un esclave Modbus doit être comprise entre 1 et 247. Le système ne vous permettra pas d'ajouter une valeur supérieure à 247.

## **⚠ AVERTISSEMENT**

### **UTILISATION D'ADRESSES MODBUS RESERVEES**

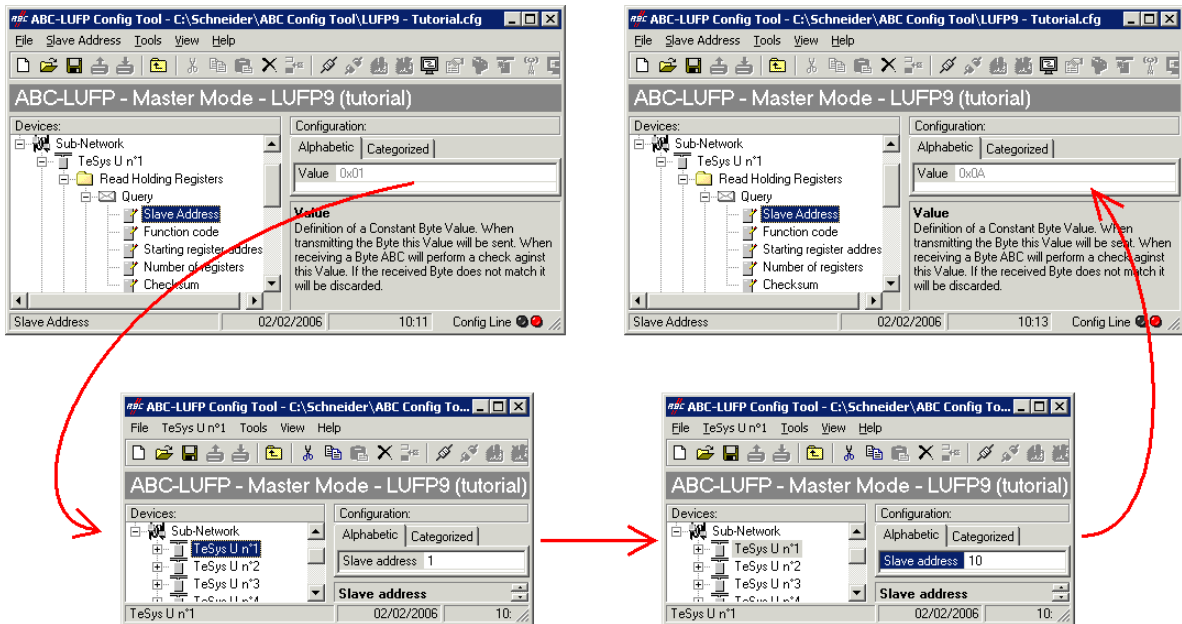
N'utilisez pas les adresses Modbus 65, 126 ou 127 si les esclaves Modbus d'une passerelle comportent un système de variation de vitesse Schneider Electric, tel qu'un démarreur Altistart ou un variateur Altivar. Les périphériques Altistart et Altivar réservent ces adresses pour d'autres communications et l'utilisation de ces adresses dans un tel système peut avoir des conséquences imprévues.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**



## 6. Configuration de la passerelle

Après validation de la nouvelle adresse (touche "Entrée" ou clic en dehors du champ de saisie de l'adresse de l'esclave Modbus), celle-ci sera prise en compte par ABC-LUFP Config Tool, et les valeurs des éléments « Slave Address » des requêtes et des réponses des commandes Modbus du nœud sélectionné seront automatiquement mises à jour. Un exemple avec la mise à jour d'un seul élément « Slave Address » est représenté ci-dessous :



### 6.11.3. Modification du nom d'une commande ou d'une transaction Modbus

Pour renommer une commande ou une transaction Modbus, vous devez d'abord effectuer l'une des opérations suivantes :

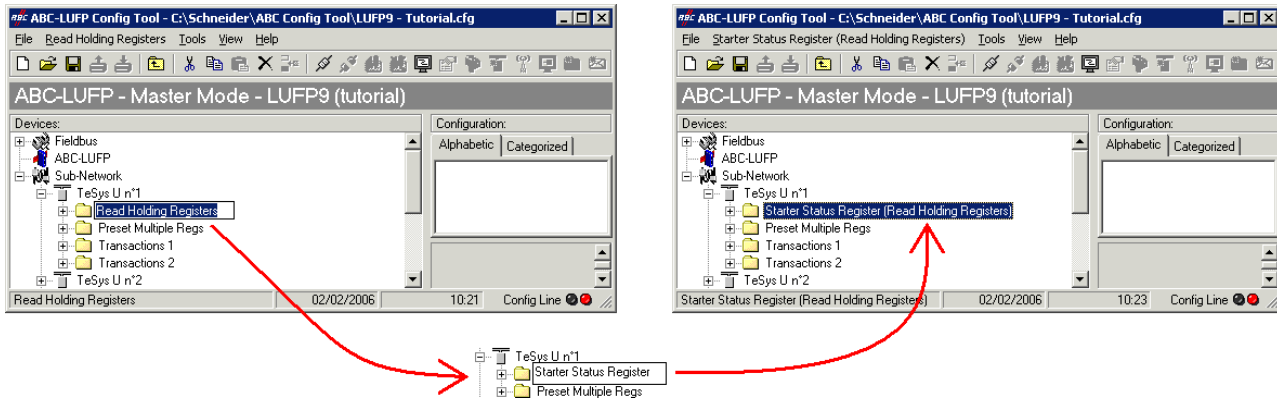
- cliquez avec le bouton droit de la souris sur le nom de la commande (ex. : Preset Multiple Regs), puis cliquez sur « Rename » dans le menu contextuel qui s'affiche, **ou**
- sélectionnez le nom de la commande, puis cliquez sur « Rename » dans le menu correspondant, **ou**
- sélectionnez le nom de la commande, puis cliquez à l'intérieur du nom, **ou**
- sélectionnez le nom de la commande, puis appuyez sur la touche F2.

Entrez ensuite un nouveau nom de commande, puis confirmez-le (touche « Entrée » ou clic en dehors du champ de nom) ou annulez-le (touche « Echap »). Une fois confirmé, le nouveau nom sera pris en compte par ABC-LUFP Config Tool.

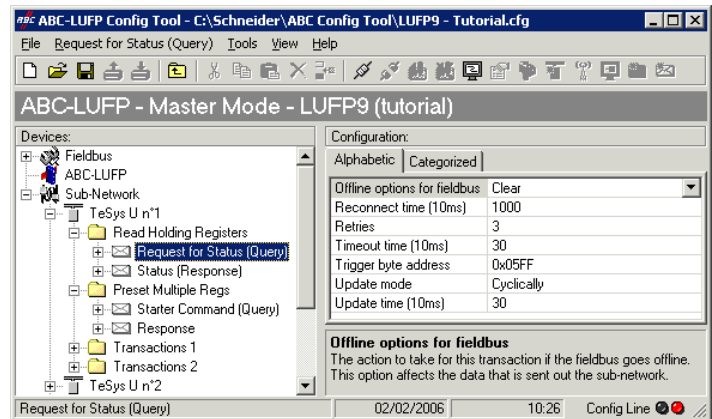
Pour les commandes Modbus, mais pas pour les transactions, le type de commande est automatiquement annexé à la fin du nouveau nom.

Un exemple est donné ci-après :

## 6. Configuration de la passerelle



Cette fonction de changement de nom peut également s'appliquer aux requêtes et aux réponses des commandes et des transactions Modbus, comme illustré ci-contre.



### 6.12. Ajout et paramétrage d'une commande Modbus

#### 6.12.1. Cas des départs-moteurs TeSys U

Dans le cas des départs-moteurs TeSys U, l'ajout d'une commande Modbus permet de commander ou de surveiller des registres supplémentaires, sans modifier la configuration par défaut. Ainsi, le fonctionnement du service périodique et des services aperiodiques de communication reste le même que celui de la configuration par défaut, contrairement aux opérations décrites dans le chapitre 6.9.

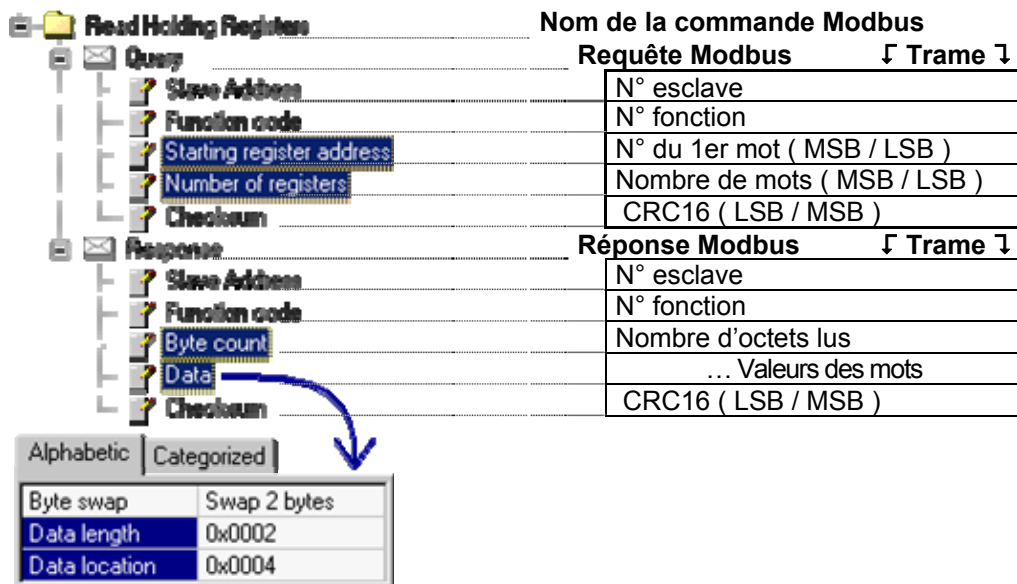
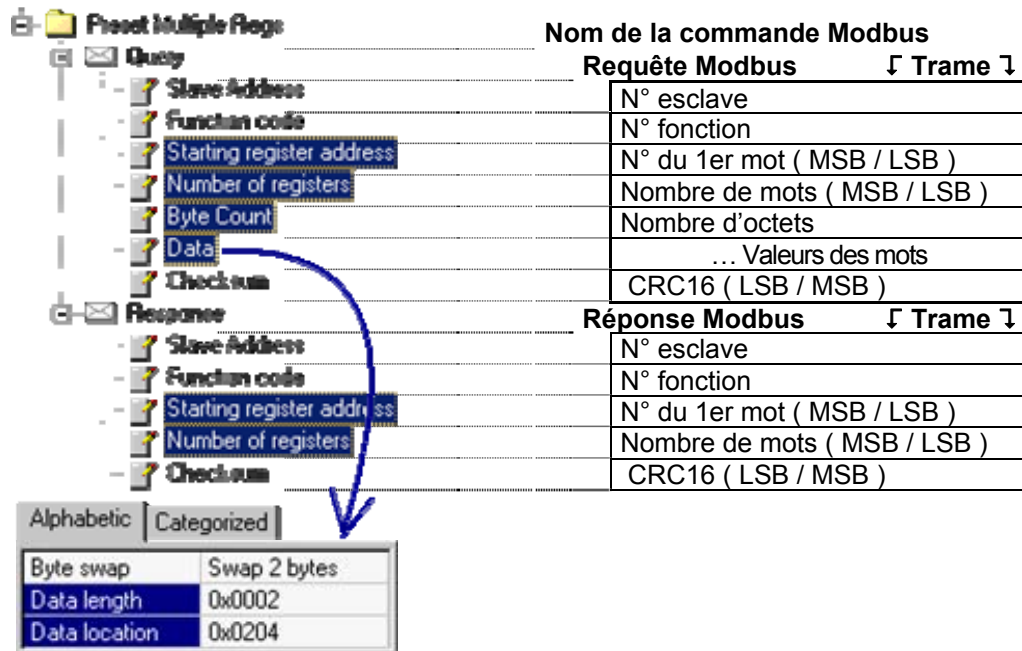
Plutôt que d'ajouter une commande et de la configurer entièrement, il est préférable de copier l'une des deux commandes par défaut « Read Holding Registers » ou « Preset Multiple Registers » à partir d'un nœud existant et de la coller dans la liste des commandes Modbus du nœud approprié.

Pour copier une commande Modbus déjà configurée à partir d'un nœud existant, sélectionnez-la, puis exécutez la commande « Copy » du menu dont le nom correspond à celui du nœud sélectionné. **Raccourci clavier :** « Ctrl C ». Continuez ensuite selon l'une des deux méthodes présentées ci-dessous :

- Sélectionnez le nœud correspondant à l'esclave Modbus pour lequel vous souhaitez ajouter cette commande (ex. : « TeSys U n°4 »), puis exécutez la commande « Paste » du menu dont le nom correspond au nœud sélectionné. Une nouvelle commande est ajoutée à la suite de toutes les autres commandes configurées pour ce nœud. L'ensemble de sa configuration est identique à celle de la commande précédemment copiée. **Raccourci clavier :** « Ctrl V ».
- Sélectionnez l'une des commandes du nœud concerné, puis exécutez la commande « Insert » du menu dont le nom correspond au nom de la commande sélectionnée. Une nouvelle commande est ajoutée juste avant celle qui est sélectionnée. L'ensemble de sa configuration est identique à celle de la commande précédemment copiée.

## 6. Configuration de la passerelle

La nouvelle commande Modbus et la commande Modbus d'origine étant identiques, vous devrez procéder aux modifications des champs **surlignés en bleu** dans l'un des deux schémas suivants, selon qu'il s'agit d'une commande « Preset Multiple Regs » ou d'une commande « Read Holding Registers » (voir chapitre 6.9). La correspondance entre les différents éléments apparaissant dans ces arborescences et la terminologie standard Modbus est située à leur droite :



**REMARQUE :** Dans tous les cas, les éléments « Query / Slave Address » et « Response / Slave Address » sont automatiquement mis à jour par ABC-LUFP Config Tool en fonction du nœud dans lequel la commande est située. Leurs valeurs ne peuvent pas être modifiées par l'utilisateur. De même, les champs « Query / Function code » et « Response / Function code » dépendent de la nature de la commande Modbus et ne peuvent pas être modifiés par l'utilisateur.

## 6. Configuration de la passerelle

---

Les opérations à effectuer sont sensiblement les mêmes que celles qui consistent à modifier les commandes par défaut. Pour la commande « Read Holding Registers », reportez-vous aux chapitres 6.9.1 et 6.9.3. Pour la commande « Preset Multiple Regs », reportez-vous aux chapitres 6.9.2 et 6.9.4.

### 6.12.2. Cas d'un esclave Modbus générique

Dans ce chapitre, nous allons ajouter et configurer des commandes Modbus différentes des commandes par défaut de la passerelle LUF9.

Reportez-vous à l'Annexe E : Commandes Modbus, pour afficher (ou obtenir) la liste des fonctions Modbus prises en charge par la passerelle LUF9. Si vous avez besoin d'utiliser une commande qui n'est pas prise en charge par la passerelle, vous pouvez la configurer. Une telle commande est englobée dans un élément spécifique appelé « Transactions » ou devient une nouvelle commande Modbus à part entière. Reportez-vous au dernier paragraphe pour plus d'informations à ce sujet.

Pour cet exemple, nous allons utiliser un démarreur Altistart, l'ATS48, et une commande Modbus reconnue par la passerelle et l'ATS48. Il s'agit de la commande « Preset Single Register », dont le code fonction est 6 et qui permet d'écrire la valeur d'un unique mot de sortie. Cette fonction servira à écrire de manière périodique la valeur du registre de commande CMD de l'ATS48, situé à l'adresse W400 (adresse 400 = 0x0190).

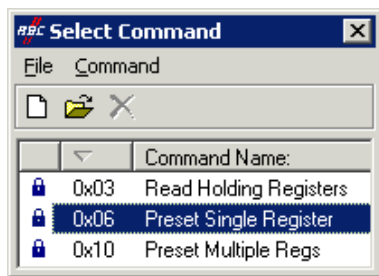
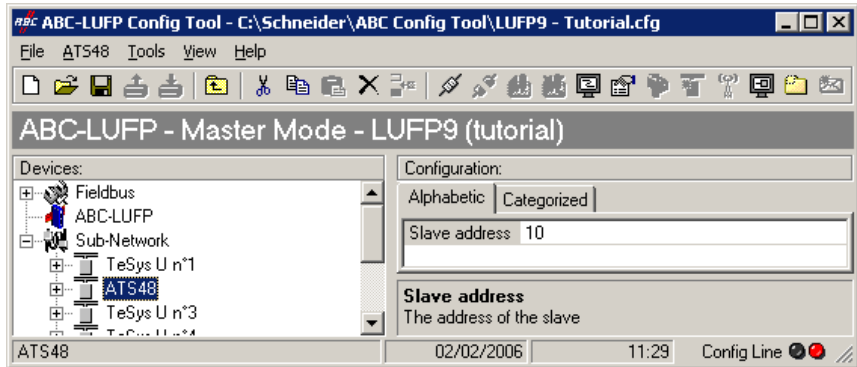
Puisque la configuration par défaut de la passerelle comporte déjà 8 esclaves Modbus, il est nécessaire d'en supprimer un, tel que le nœud « TeSys U n°2 », par exemple, et d'ajouter un nouveau nœud à sa place (voir les chapitres 6.7 et 6.8).

**REMARQUE** : Il est fortement déconseillé de supprimer le nœud « TeSys U n°1 », car il contient les commandes qui correspondent aux services de lecture et d'écriture d'un paramètre dans un esclave Modbus.

## 6. Configuration de la passerelle

Après avoir créé le nouveau nœud, nous devons le renommer et lui attribuer l'adresse Modbus 10, comme indiqué ci-contre :

Nous ajoutons ensuite la commande « Preset Single Register » en exécutant la commande « Add Command » du menu « ATS48 ».



Dans la fenêtre qui apparaît (reproduite ci-contre), sélectionnez la commande « 0x06 Preset Single Register » et exécutez la commande « Select » du menu « File ».

De retour dans la fenêtre principale de ABC-LUFP Config Tool, la commande « Preset Single Register » apparaît désormais dans la liste des commandes Modbus du nœud « ATS48 ».

Développez l'arborescence complète de cette commande, comme illustré ci-dessous. La correspondance entre les différents éléments apparaissant dans cette arborescence et la terminologie standard Modbus est située à sa droite.

ATS48		Nom de l'esclave Modbus
Preset Single Register		Nom de la commande Modbus
Query		Requête Modbus
		↓ Trame ↓
Slave Address		N° esclave
Function code		N° fonction
Register address		N° du mot ( MSB / LSB )
Preset data		Valeur du mot ( MSB / LSB )
Checksum		CRC16 ( LSB / MSB )
Response		Réponse Modbus
		↓ Trame ↓
Slave Address		N° esclave
Function code		N° fonction
Register address		N° du mot ( MSB / LSB )
Preset data		Valeur du mot ( MSB / LSB )
Checksum		CRC16 ( LSB / MSB )

Ces éléments peuvent être configurés à l'aide de ABC-LUFP Config Tool, comme décrit dans les chapitres suivants.

## 6. Configuration de la passerelle

### 6.12.2.1. Gestion des modes dégradés

#### Arrêt ou défaillance du processeur de l'automate

<b>Réponse du processeur de l'automate</b>
Sorties : Erreur logicielle, réinitialisation des sorties sur leur état par défaut ou conservation de leur état actuel, selon la configuration. Erreur matérielle (EEPROM ou défaillance matérielle), état de sortie indéterminé. Entrées : L'automate cesse de répondre aux entrées quel que soit l'état d'erreur.
<b>Réponse du scanner DeviceNet</b>
En fonction de la configuration du scanner : le scanner cesse de communiquer avec la passerelle LUF9, force les sorties DeviceNet sur la valeur 0 et actualise les entrées, ou maintient les sorties DeviceNet sur leur dernière position et actualise les entrées.
<b>Réponse de la passerelle LUF9</b>
Si le scanner cesse de communiquer avec la passerelle, le comportement dépend des options « Offline options for fieldbus » : Clear : Toutes les données envoyées à l'esclave Modbus concerné ont la valeur 0. Freeze : Toutes les données envoyées conservent leur valeur actuelle. No scanning : La requête n'est plus transmise.
Si le scanner force les sorties DeviceNet sur la valeur 0 et actualise les entrées : toutes les données émises (requêtes d'écriture) sont configurées sur la valeur 0, la lecture à partir des esclaves continue de s'effectuer normalement.
Si le scanner maintient les sorties DeviceNet et actualise les entrées : toutes les données envoyées (requêtes d'écriture) conservent leur valeur actuelle, la lecture à partir des esclaves continue de s'effectuer normalement.
<b>Réponse de l'esclave</b>
La réponse dépend de chaque esclave.

#### Arrêt ou défaillance du scanner DeviceNet

<b>Réponse du processeur de l'automate</b>
Le processeur de l'automate fournit à l'application plusieurs erreurs et/ou objets de diagnostic au cas où le scanner DeviceNet cesserait de fonctionner ou connaîtrait une défaillance (entrée/sortie non valide). Reportez-vous au manuel d'utilisation de l'automate pour consulter leur description. Ces informations doivent être gérées dans l'application de l'automate.
<b>Réponse du scanner DeviceNet</b>
Si le scanner DeviceNet est arrêté (commande provenant de l'application) : le scanner cesse de communiquer avec la passerelle LUF9.
Si le scanner DeviceNet connaît une défaillance, le scanner cesse de communiquer avec le processeur et la passerelle LUF9.
<b>Réponse de la passerelle LUF9</b>
Si le scanner cesse de communiquer avec la passerelle, le comportement dépend des options « Offline options for fieldbus » : Clear : Toutes les données envoyées à l'esclave Modbus concerné ont la valeur 0. Freeze : Toutes les données envoyées conservent leur valeur actuelle. No scanning : La requête n'est plus transmise.
<b>Réponse de l'esclave</b>
La réponse dépend de chaque esclave.

## 6. Configuration de la passerelle

---

### Passerelles LUF9P9 déconnectées du côté DeviceNet

<b>Réponse de l'automate</b>
Le processeur de l'automate fournit plusieurs objets d'erreur et de diagnostic provenant du scanner DeviceNet en cas de déconnexion d'un esclave de l'application. Reportez-vous au manuel d'utilisation de l'automate pour consulter leur description. Ces informations doivent être gérées dans l'application de l'automate.
<b>Réponse du scanner DeviceNet</b>
Le scanner DeviceNet fournit au processeur différents objets d'erreur et de diagnostic en cas de déconnexion d'un esclave DeviceNet.
<b>Réponse de la passerelle LUF9P9</b>
Le comportement dépend des options « Offline options for fieldbus » : Clear : Toutes les données envoyées à l'esclave Modbus concerné ont la valeur 0. Freeze : Toutes les données envoyées conservent leur valeur actuelle. No scanning : La requête n'est plus transmise.
<b>Réponse de l'esclave</b>
La réponse dépend de chaque esclave.

### Défaillance des passerelles LUF9P9

<b>Réponse de l'automate</b>
Le processeur de l'automate fournit plusieurs objets d'erreur et de diagnostic provenant du scanner DeviceNet en cas de défaillance d'un esclave vers l'application. Reportez-vous au manuel d'utilisation de l'automate pour consulter leur description. Ces informations doivent être gérées dans l'application de l'automate.
<b>Réponse du scanner DeviceNet</b>
Le scanner DeviceNet fournit au processeur différents objets d'erreur et de diagnostic en cas de défaillance d'un esclave DeviceNet.
<b>Réponse de la passerelle LUF9P9</b>
En cas de défaillance, la passerelle cesse de communiquer avec le scanner DeviceNet et les esclaves Modbus.
<b>Réponse de l'esclave</b>
La réponse dépend de chaque esclave.

## 6. Configuration de la passerelle

---

### Passerelles LUF99 déconnectées du côté Modbus ou défaillance d'un esclave

<b>Réponse de l'automate</b>
Le processeur donne accès au mot d'état de la passerelle provenant de la table d'entrée du scanner DeviceNet, ainsi qu'au mot de commande de la passerelle provenant de la table de sortie. Ces 2 mots doivent être gérés dans l'application de l'automate afin de détecter si un esclave Modbus est manquant.
<b>Réponse du scanner DeviceNet</b>
Le scanner DeviceNet doit être configuré de façon à accéder à l'état de la passerelle et aux mots de commande afin de fournir des informations de diagnostic Modbus.
<b>Réponse de la passerelle LUF99</b>
Le comportement dépend des différentes options : Timeout time, nombre de Retries, Reconnect time et Offline option for sub-network.
<b>Réponse de l'esclave</b>
En cas de déconnexion Modbus, le comportement dépend de chaque esclave. En cas de défaillance d'un esclave, il présente un état indéterminé qui doit être géré dans l'application de l'automate.



## 6. Configuration de la passerelle


### 6.12.2.2. Configuration de la requête

Sélectionnez l'élément « Query » de la commande Modbus. Les différents éléments de la configuration de la requête de cette commande sont reproduits ci-contre. Les valeurs affichées correspondent aux valeurs par défaut pour toute nouvelle commande.

Ces éléments permettent de configurer la gestion de l'ensemble de la commande, y compris la gestion des modes dégradés (nombre de ré-émissions, par exemple).

Alphabetic	Categorized
Offline options for fieldbus	Clear
Reconnect time (10ms)	1000
Retries	3
Timeout time (10ms)	100
Trigger byte address	0x05FF
Update mode	Cyclically
Update time (10ms)	100

Chacun de ces éléments est décrit, dans l'ordre, dans le tableau situé ci-dessous. Lorsqu'une unité est attribuée à un élément, elle est indiquée entre parenthèses à la suite du nom de l'élément :

Élément de configuration	Description
Offline options for fieldbus	<p>Cet élément affecte les données <b>envoyées à l'esclave Modbus</b>, et pour la seule requête à laquelle appartient cet élément, lorsque la passerelle est déconnectée du réseau DeviceNet. Cet élément prend l'une des trois valeurs suivantes :</p> <ul style="list-style-type: none"> <li>- Clear ..... Les données envoyées à l'esclave Modbus à l'aide de cette requête sont désormais égales à 0x0000 (RAZ des données de sortie dans la mémoire de la passerelle).</li> <li>- Freeze ..... Les données envoyées à l'esclave Modbus à l'aide de cette requête conservent leur valeur actuelle (gel des données de sortie dans la mémoire de la passerelle).</li> <li>- NoScanning ... La requête n'est plus envoyée à l'esclave Modbus par la passerelle.</li> </ul>
Reconnect time (10ms)  Valeur par défaut : 10ms x 1000 = 10s	<p>En cas de non-réponse de l'esclave Modbus à une requête, ou suite à la réception d'une réponse erronée, la passerelle utilise les éléments « Retries » et « Timeout time (10ms) » pour effectuer des ré-émissions. Si l'esclave Modbus n'a toujours pas répondu correctement suite à ces ré-émissions, la passerelle cesse de lui envoyer la requête correspondante pendant une durée réglable à l'aide de l'élément « Reconnect time (10ms) ».</p> <p>Lorsque cette durée s'achève, la passerelle tente de restaurer la communication avec l'esclave Modbus.</p>
Retries  Valeur par défaut : 3	<p>Cet élément indique le nombre de ré-émissions effectuées par la passerelle en cas de non-réponse de l'esclave Modbus à une requête, ou en cas de réponse erronée. Ce processus de ré-émission cesse dès que la passerelle obtient dans les temps une réponse correcte. Si aucune des ré-émissions n'a permis à la passerelle d'obtenir une réponse correcte, l'esclave Modbus est considéré comme étant déconnecté, mais uniquement vis-à-vis de la commande concernée. La passerelle utilise alors les éléments « Offline options for sub-network » et « Reconnect time (10ms) » et la DEL  MODBUS devient rouge. Celle-ci ne redeviendra verte qu'à condition que la commande Modbus associée obtienne une réponse correcte, suite à la reprise des communications (voir élément « Reconnect time (10ms) »).</p> <p>Si le nombre de ré-émissions est égal à 0, le processus décrit ci-dessus ne sera pas exécuté.</p>

## 6. Configuration de la passerelle

Élément de configuration	Description
Timeout time (10ms)  Valeur par défaut : 10ms x 100 = 1s	<p>Cet élément représente le temps d'attente d'une réponse de la part de la passerelle. Si une réponse n'est pas parvenue à la passerelle dans le temps imparti, configuré à l'aide de l'élément « timeout time (10ms) », la passerelle procède à une ré-émission. Ce processus continue jusqu'à atteindre la dernière ré-émission autorisée (voir élément « Retries »), puis la passerelle déclare l'esclave Modbus comme étant déconnecté, mais uniquement vis-à-vis de la commande à laquelle appartient l'élément « timeout time (10ms) ».</p>
Trigger byte address	<p>Cet élément n'est utilisé par la passerelle qu'à la condition que « Update mode » soit égal à « Change of state on trigger ». Dans ce cas, il spécifie l'adresse, dans la mémoire de sortie de la passerelle (0x0202 à 0x03FF), d'un compteur 8 bits géré par le maître DeviceNet.</p> <p>Lorsque le maître DeviceNet actualise la valeur de l'élément « Trigger Byte Address » en lui attribuant toute autre valeur que zéro, la requête configurée à l'aide d'un « Update Mode » défini sur « Change of state on trigger » est transmise à l'esclave Modbus. Le maître DeviceNet doit donc avoir accès à ce compteur de la même manière que pour les registres de sortie périodiques envoyés aux départs-moteurs TeSys U.</p> <p>Comparativement à la valeur « On data change » du « Update Mode », ce mode permet d'envoyer une commande sur ordre spécifique du maître DeviceNet si, par exemple, celui-ci ne peut pas mettre à jour l'ensemble des données d'une requête au même moment.</p> <p><b>REMARQUE</b> : Dans le cas de la configuration par défaut de la passerelle, le mode des commandes personnalisées « Transactions 1 » et « Transactions 2 » du nœud « TeSys U n°1 » est défini sur « Change of state on trigger ». Ces commandes aperiodiques servent, respectivement, à lire et à écrire la valeur d'un paramètre de l'un des esclaves Modbus. Les éléments « Trigger byte address » des éléments « Query » de ces deux commandes sont configurés aux adresses 0x021E et 0x021F. Il s'agit des « compteurs de requête de la lecture/écriture d'un paramètre ». Vis-à-vis du scanner DeviceNet et de RSNetWorx, ces deux données sont configurées de la même manière que les autres sorties (voir chapitre 4.2.4) et correspondent à la sortie O:1.16.</p> <p>Pour émettre l'une de ces deux commandes, l'automate maître DeviceNet devra tout d'abord mettre à jour l'ensemble des données à transmettre sur le réseau Modbus pour cette commande (adresses 0x0212 à 0x0217 ou adresses 0x0218 à 0x021D), puis modifier la valeur du compteur associé (adresse 0x021E ou 0x021F). La passerelle transmettra alors la requête qui correspond à la commande.</p> <p><b>REMARQUE</b> : Il n'est pas obligatoire que le « trigger byte » soit une donnée de sortie mise à jour par le maître DeviceNet. Il est tout à fait envisageable qu'il s'agisse d'une entrée comprise entre 0x0002 et 0x01FF. Dans ce cas, l'esclave Modbus qui met à jour cet octet conditionnera les échanges de la commande en cours de configuration.</p>

## 6. Configuration de la passerelle

Elément de configuration	Description
Update mode	<p>Cet élément sert à préciser le mode d'émission de la requête sur le réseau Modbus. Il prend l'une des quatre valeurs suivantes :</p> <ul style="list-style-type: none"> <li>- Cyclically..... Mode de communication par défaut. La requête est transmise de manière <b>périodique</b> sur le réseau Modbus (voir élément « Update time »).</li> <li>- On data change..... La passerelle transmet la requête sur le réseau Modbus lorsqu'au moins l'une des données de cette requête est modifiée par le maître DeviceNet. Il s'agit donc d'un mode de communication <b>apériodique</b>.</li> <li>- Single Shot..... Ce mode de transmission n'autorise qu'un seul échange Modbus pour toute la durée de fonctionnement de la passerelle. Cet échange a lieu juste après l'initialisation de celle-ci.</li> <li>- Change of state on trigger..... Avec ce mode de communication <b>apériodique</b>, la requête Modbus est envoyée à chaque fois que le maître DeviceNet modifie la valeur d'un compteur 8 bits désigné par l'élément « Trigger byte address ». C'est le cas, par exemple, des requêtes associées aux commandes personnalisées « Transactions 1 » et « Transactions 2 » du nœud « TeSys U n °1 » de la configuration par défaut de la passerelle. Ces requêtes sont transmises lorsque la valeur des « trigger bytes » associés (adresses 0x021E et 0x021F) sont modifiées par le maître DeviceNet. Reportez-vous à la description de cet élément pour obtenir de plus amples informations sur l'utilité de ce mode de communication.</li> </ul>
Update time (10ms)  Valeur par défaut : 10ms x 100 = 1s	<p>Cet élément n'est utilisé par la passerelle qu'à la condition que « Update mode » soit défini sur « Cyclically ». Dans ce cas, il spécifie la période d'émission de la requête sur le réseau Modbus.</p>

Pour revenir à notre exemple utilisant l'ATS48 à l'adresse 10, nous utiliserons la configuration présentée ci-contre. Les points notables de cette configuration sont les suivants :

- Lors de la déconnexion, les données sont réinitialisées sur les deux réseaux.
- 3 ré-émissions avec un délai timeout de 100 ms.
- Les communications périodiques ont un « Update time » cyclique égal à 300 ms.

Alphabetic	Categorized
Offline options for fieldbus	Clear
Reconnect time (10ms)	1000
Retries	3
Timeout time (10ms)	10
Trigger byte address	0x05FF
Update mode	Cyclically
Update time (10ms)	30

### 6.12.2.3. Configuration de la réponse

Sélectionnez ensuite l'élément « Response » de la commande Modbus. Les différents éléments de la configuration de la réponse à cette commande sont reproduits ci-contre. Les valeurs affichées correspondent aux valeurs par défaut pour toute nouvelle commande.

Alphabetic	Categorized
Offline options for sub-network	Clear
Trigger byte	Disabled
Trigger byte address	0x05FF

Ces éléments permettent de configurer un seul aspect de la gestion de la commande, décrit en haut de la page de droite. Chacun de ces éléments est décrit, dans l'ordre, dans le tableau situé ci-dessous.

## 6. Configuration de la passerelle

Elément de configuration	Description
Offline options for sub-network	<p>Cet élément affecte les données d'entrée <b>envoyées au maître Modbus</b>, mais uniquement les données de la réponse à laquelle appartient cet élément, chaque fois que l'esclave Modbus ne répond pas à la requête correspondante (ou en cas de déconnexion du sous-réseau Modbus).</p> <p>Cet élément prend l'une des deux valeurs suivantes :</p> <ul style="list-style-type: none"> <li>- Clear Toutes les données envoyées au maître DeviceNet pour cette réponse sont égales à 0x0000 (RAZ des données d'entrée dans la mémoire de la passerelle).</li> <li>- Freeze Toutes les données envoyées au maître DeviceNet pour cette réponse conservent leur valeur actuelle (gel des données d'entrée dans la mémoire de la passerelle).</li> </ul>
Trigger byte	<p>Cet élément est utilisé par la passerelle pour activer ou non l'incrémement unitaire d'un compteur 8 bits afin de signaler au maître DeviceNet la réception d'une nouvelle réponse à la commande Modbus associée. Il prend l'une des deux valeurs suivantes :</p> <ul style="list-style-type: none"> <li>- Disabled..... Configuration par défaut. La passerelle n'incrémente aucun compteur sur réception de la réponse Modbus.</li> <li>- Enabled..... Chaque fois que la passerelle reçoit une nouvelle réponse à la commande Modbus associée, elle incrémente la valeur d'un compteur 8 bits désigné par l'élément « Trigger byte address » (voir ci-dessous). Cette modification de la valeur de l'élément « Trigger Byte Address » peut être utilisée pour indiquer au maître DeviceNet que les données de la réponse Modbus sont déjà sur le point d'être interrogées.</li> </ul>
Trigger byte address	<p>Cet élément n'est utilisé par la passerelle qu'à la condition que l'élément « Trigger byte » soit défini sur « Enabled ». Dans ce cas, il spécifie l'adresse, dans la mémoire d'entrée de la passerelle (0x0002 0 0x01FF), d'un compteur 8 bits géré par la passerelle.</p> <p>Lorsque la passerelle reçoit une réponse à la commande Modbus associée, elle incrémente la valeur de ce compteur de manière unitaire (valeur = valeur+1). Le maître DeviceNet doit donc avoir accès à ce compteur de la même manière que pour les registres d'entrée périodiques issus des départs-moteurs TeSys U.</p> <p>Ce mode permet d'informer le maître DeviceNet qu'une nouvelle réponse est disponible. Cela peut être utile, par exemple, s'il est possible que les données de deux réponses successives soient identiques.</p> <p><b>REMARQUE</b> : Dans le cas de la configuration par défaut de la passerelle, l'élément « Trigger byte » des réponses aux commandes personnalisées « Transactions 1 » et « Transactions 2 » du nœud « TeSys U n°1 » est défini sur « Enabled ». La gestion des réponses aux commandes de lecture et d'écriture de paramètres est donc <b>événementielle</b>.</p> <p>Les éléments « Trigger byte address » des éléments « Response » de ces deux commandes sont configurés aux adresses 0x001E et 0x001F. Il s'agit des « compteurs de réponse de la lecture/écriture d'un paramètre ». Vis-à-vis du scanner DeviceNet et de RSNetWorx, ces deux données sont configurées de la même manière que les autres entrées (voir chapitre 4.2.6) et correspondent à l'entrée I:1.16.</p> <p>L'automate maître DeviceNet pourra détecter la réception d'une réponse de la part d'un esclave Modbus en comparant la valeur précédente et la valeur actuelle du compteur associé (adresse 0x001E or 0x001F). S'il y a eu <i>incrémement unitaire</i> de ce compteur, l'automate pourra, par exemple, lire l'ensemble des données de la réponse (adresses 0x0013 à 0x0017 ou adresses 0x0018 à 0x001D) et autoriser l'émission d'une nouvelle requête de lecture ou d'écriture de la valeur d'un paramètre (à l'aide d'un « Trigger byte » dédié aux requêtes). Contrairement aux autres compteurs « Query », la valeur enregistrée à l'adresse « Response » du Trigger byte est un véritable compteur 256, <i>c'est-à-dire</i> que la valeur nulle doit être prise en compte (... 254, 255, 0, 1, 2 ...).</p>

Dans cet exemple utilisant l'ATS48, nous ne souhaitons pas que la réponse devienne événementielle. Nous conserverons par conséquent la configuration par défaut.

## 6. Configuration de la passerelle

### 6.12.2.4. Configuration du contenu de la trame de la requête

La fenêtre reproduite ci-dessous est obtenue à l'aide de la commande « Edit Transaction » du menu « Query ». Contrairement à l'arborescence de la fenêtre principale de ABC-LUFP Config Tool, cet affichage présente l'avantage de visualiser l'ensemble des champs de la trame en même temps que leurs valeurs. Les valeurs affichées ci-dessous correspondent aux valeurs affectées par défaut à la requête de la commande Modbus que nous avons créée. Sous cette fenêtre a été ajoutée la correspondance avec le contenu de la trame Modbus correspondante.

Slave Address	Function code	Register address	Preset data			Checksum	
Value	Value	Value	Data location	Data length	Byte swap	Error check type	Error check start byte
0x0A	0x06	0x0000	0x0202	0x0002	No swapping	CRC	0x0000
N° esclave	N° fonction	Numéro du mot (MSB / LSB)	Valeur du mot (MSB / LSB)			CRC16 (LSB / MSB)	

Editez les valeurs non grisées les unes après les autres. Leur description est fournie ci-après.

La nature des champs d'une trame dépend de la commande Modbus à laquelle elle correspond. Cependant, un certain nombre de ces champs sont communs à toutes les trames, tandis que d'autres sont communs à plusieurs d'entre elles. Voici la description de celles qui sont présentées ci-dessus, dans le cadre de l'exemple décrit au début du chapitre 6.12.2 :

Champ dans la trame	Taille dans la trame	Description
Slave Address	1 octet	Ce champ ne peut pas être modifié par l'utilisateur et sa valeur est grisée pour le lui signaler. ABC-LUFP Config Tool met à jour la valeur de ce champ de manière automatique à l'aide de l'adresse de l'esclave Modbus qui correspond au nœud courant. <b>REMARQUE</b> : Ce champ est commun aux requêtes de toutes les commandes Modbus. Exemple : La valeur de ce champ est définie sur l'adresse de l'esclave Modbus qui correspond aux nœuds « ATS48 », c'est-à-dire à 0x0A.
Function code	1 octet	Ce champ ne peut pas être modifié par l'utilisateur et sa valeur est grisée pour le lui signaler. ABC-LUFP Config Tool met à jour la valeur de ce champ de manière automatique à l'aide du code fonction de la commande Modbus correspondante. <b>REMARQUE</b> : Ce champ est commun aux requêtes de toutes les commandes Modbus. Exemple : La valeur de ce champ est égale au code de la commande « Preset Single Register » (écriture de la valeur d'un mot de sortie), c'est-à-dire à 0x06.
Register address	2 octets	Adresse d'un mot de sortie, ou d'un registre, dans la mémoire de l'esclave Modbus. Ce champ désigne donc l'objet mémoire sur lequel porte la commande. <b>REMARQUE</b> : Ce champ est commun aux requêtes de toutes les commandes Modbus ayant pour but d'accéder à un ou plusieurs emplacements dans la mémoire d'un esclave Modbus. Dans le cas d'un accès à plusieurs emplacements mémoire, le champ « Register » désigne l'adresse du premier mot pris pour objet par la commande. Exemple : La valeur de ce champ doit être modifiée en saisissant l'adresse du registre de commande CMD, c'est-à-dire 400 (0x0190). Cette valeur sera automatiquement convertie au format hexadécimal si l'utilisateur la saisit au format décimal.

## 6. Configuration de la passerelle

Champ dans la trame	Taille dans la trame	Description
Preset Data	2 octets ou plus s'il s'agit d'un bloc de données	<p><u>Data Location</u> : Adresse, dans la mémoire des données de sortie de la passerelle (0x0202 à 0x03FF), de la donnée à transmettre dans le champ « Preset Data » de la trame de la requête.</p> <p><b>REMARQUE</b> : Le champ « Data location » est utilisé pour chaque trame permettant de faire transiter des données entre les esclaves Modbus et le maître DeviceNet. Dans ce cas, il désigne l'adresse de début du bloc de données à transmettre.</p> <p>Lorsque vous sélectionnez une valeur pour le champ « Data Location », les données doivent être situées à des adresses paires afin d'aligner les données Modbus (au format 16 bits) sur les sorties O:1.x du scanner DeviceNet. Si les données ne sont pas situées à des adresses paires, les valeurs prévues pour les registres Modbus peuvent être réparties sur deux mots de l'automate DeviceNet. Ceci complique considérablement la programmation de l'application, car celle-ci peut être contrainte d'analyser un mot de l'automate pour l'octet Modbus LSB et un autre pour l'octet Modbus MSB. Si ce problème n'est pas géré correctement, il est possible de lire et d'écrire des valeurs de données erronées sur les esclaves Modbus.</p>

### AVERTISSEMENT

#### RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

L'utilisateur doit utiliser des valeurs paires pour le champ « Data Location » (ex. : 514, 516, 518, etc.). La sélection d'emplacements de données impaires complique la programmation de l'application et augmente les risques d'écriture ou de lecture de valeurs Modbus incorrectes sur ou depuis les périphériques esclaves. Selon la configuration de l'utilisateur, cela peut provoquer un fonctionnement imprévu de l'appareil.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

		<p>Pour revenir à notre exemple précédent, la valeur à affecter au registre CMD de l'ATS48 doit être placée dans la zone des données de sortie de la passerelle. Nous utiliserons le premier emplacement libre commençant à une adresse paire, c'est-à-dire celui qui est situé à l'adresse 0x0220, dans le cas de la configuration par défaut de la passerelle.</p> <p><u>Data length</u> : Longueur du bloc des données de sortie, dans la mémoire de la passerelle, dont les valeurs doivent être transmises dans le champ « Preset Data » de la trame de la requête. Elle est exprimée en nombre d'octets.</p> <p><b>REMARQUE</b> : Le champ « Data length » est toujours utilisé conjointement au champ « Data location », décrit ci-dessus.</p> <p>Exemple : Puisque la commande « Preset Single Register » sert à écrire la valeur d'un seul registre (16 bits), la valeur du champ « Data length » doit être égale à 2.</p> <p>Consultez la documentation de chaque esclave Modbus pour connaître le nombre maximum de données 8 bits qu'il est possible de placer dans les champs de type « Data » des requêtes et des réponses de cet esclave. Dans le cas de l'ATS48, par exemple, ce nombre est limité à 30 mots de 16 bits (la longueur du champ « Data » est limitée à ≤ 60).</p>
--	--	---

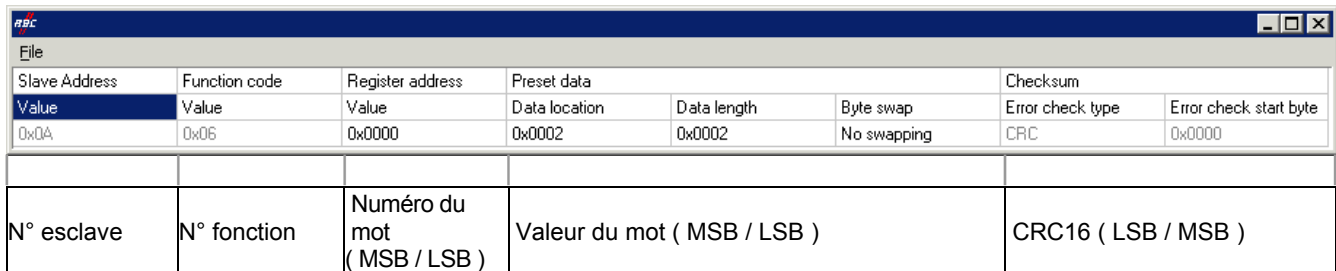
## 6. Configuration de la passerelle

Champ dans la trame	Taille dans la trame	Description
		<p><u>Byte swap</u>: Précise si les octets des données de sortie à transmettre à l'esclave Modbus doivent être ou non permutés avant d'être placés dans la trame Modbus. Les trois valeurs possibles sont les suivantes :</p> <ul style="list-style-type: none"> <li>- No swapping ..... Configuration par défaut. Les données sont transmises dans le même ordre que celui de leur présence dans la mémoire de la passerelle.</li> <li>- Swap 2 bytes ..... Les octets à transmettre sont permutés deux à deux. Pour une donnée 16 bits, l'octet de poids fort est placé en premier dans la trame Modbus, alors qu'elle est toujours écrite dans la mémoire de la passerelle par un maître DeviceNet avec l'octet de poids faible en premier.</li> <li>- Swap 4 bytes ..... Les octets à transmettre sont permutés quatre à quatre. Ce cas est très peu utilisé, car il concerne uniquement les données 32 bits. Son principe est similaire à celui du cas précédent, « Swap 2 bytes ».</li> </ul> <p><b>REMARQUE</b> : Avec DeviceNet, utilisez « Swap 2 bytes ».</p> <p>Exemple : Nous utiliserons la valeur « Swap 2 bytes », car les deux octets de la valeur à écrire dans le registre CMD de l'ATS48, transmis par l'automate SLC500, sont placés dans la mémoire de la passerelle dans l'ordre poids faible / poids fort.</p>
Checksum	2 octets	<p><u>Error check type</u> : Type du contrôle d'erreur pour la trame.</p> <ul style="list-style-type: none"> <li>- CRC ..... Méthode par défaut.</li> </ul> <p>Il s'agit de la méthode qui a été adoptée pour le protocole Modbus RTU. Il est impossible de la changer.</p> <p><u>Error check start byte</u> : Indique le numéro de l'octet, dans la trame, à partir duquel le calcul du « checksum » doit commencer. Le premier octet de chaque trame porte le numéro 0.</p> <p><b>REMARQUE</b> : Le calcul du checksum d'une trame doit toujours commencer par le premier octet. Ne remplacez pas la valeur par défaut « zéro » de l'élément « Error check start byte ». Une valeur différente de zéro provoquera une erreur de CRC et toutes les communications Modbus retourneront une erreur.</p>

## 6. Configuration de la passerelle

### 6.12.2.5. Configuration du contenu de la trame de la réponse

La fenêtre reproduite ci-dessous est obtenue à l'aide de la commande « Edit Transaction » du menu « Response ». Les valeurs qui y sont présentées correspondent aux valeurs affectées par défaut à la réponse de la commande Modbus que nous avons créée. Sous cette fenêtre a été ajoutée la correspondance avec le contenu de la trame Modbus correspondante.



Slave Address	Function code	Register address	Preset data			Checksum	
Value	Value	Value	Data location	Data length	Byte swap	Error check type	Error check start byte
0x04	0x06	0x0000	0x0002	0x0002	No swapping	CRC	0x0000

N° esclave	N° fonction	Numéro du mot (MSB / LSB)	Valeur du mot (MSB / LSB)	CRC16 (LSB / MSB)
------------	-------------	---------------------------	---------------------------	-------------------

Editez les valeurs non grisées les unes après les autres.

Leur description est fournie sur la page suivante, mais reportez-vous également au chapitre précédent, car la nature du contenu des trames des réponses est très proche de celle des champs des trames des requêtes Modbus.

**REMARQUE** : Si la valeur de l'un des champs de la réponse d'un esclave Modbus est différente de celle qui est configurée via ABC-LUFP Config Tool, la réponse sera refusée par la passerelle. Celle-ci procédera alors à une ré-émission de la requête, à condition qu'au moins une ré-émission ait été configurée pour cette commande (voir chapitre 6.12.2.2).



## 6. Configuration de la passerelle

Champ dans la trame	Taille dans la trame	Description
Slave Address	1 octet	Identique à celle du champ « Slave Address » de la requête.
Function code	1 octet	Identique à celle du champ « Function » de la requête.
Register address	2 octets	Identique à celle du champ « Register » de la requête, puisque la réponse Modbus, dans le cas de la commande « Preset Single Register », est un écho à la requête correspondante. Vous devez ici aussi saisir l'adresse de l'objet mémoire sur lequel porte la commande. Si vous recevez un code d'exception, reportez-vous à (*).
Preset Data	2 octets ou plus s'il s'agit d'un bloc de données	<u>Data Location</u> : Adresse, dans la mémoire des données d'entrée de la passerelle (0x0002 à 0x01FF), de la donnée reçue dans le champ « Preset Data » de la trame de la réponse. <b>REMARQUE</b> : Veillez à ce que les données soient situées à des adresses paires afin d'aligner les données Modbus (au format 16 bits) sur les entrées I:1.x du scanner DeviceNet. Exemple : La valeur renvoyée en guise d'écho à la commande doit être placée dans la zone des données d'entrée de la passerelle. Nous utiliserons le premier emplacement libre, c'est-à-dire celui qui est situé à l'adresse 0x0020, dans le cas de la configuration par défaut de la passerelle. Si vous recevez un code d'exception, reportez-vous à (*).

### AVERTISSEMENT

#### RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

L'utilisateur doit utiliser des valeurs paires pour le champ « Data Location » (ex. : 2, 4, 6, etc.). La sélection d'emplacements de données impaires complique la programmation de l'application et augmente les risques d'écriture ou de lecture de valeurs Modbus incorrectes sur ou depuis les périphériques esclaves. Selon la configuration de l'utilisateur, cela peut provoquer un fonctionnement imprévu de l'appareil.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

		<p><u>Data length</u> : Longueur du bloc des données d'entrée reçues dans le champ « Preset Data » de la trame de la réponse. Elle est exprimée en nombre d'octets. Exemple : La valeur du champ « Data length » doit être égale à 2.</p> <p><u>Byte swap</u> : Identique à celle du champ « Byte swap » de la requête (reportez-vous au tableau des requêtes pour plus d'informations). Exemple : Nous utiliserons ici aussi la valeur « Swap 2 bytes », pour les mêmes raisons que dans le cas de la requête.</p>
Checksum	2 octets	<p>Error check type : Identique à celle du champ « Error check type » de la requête.</p> <p>Error check start byte : Identique à celle du champ « Error check start byte » de la requête.</p> <p><b>REMARQUE</b> : Ces deux champs ne peuvent pas être modifiés par l'utilisateur et leurs valeurs sont grisées pour le lui signaler. ABC-LUFP Config Tool met à jour les valeurs de ces champs de manière automatique à l'aide de celles des champs « Error check type » et « Error check start byte » de la requête.</p>

(\*) Si vous recevez un code d'exception, la passerelle procède à la ré-émission de la requête conformément au nombre de nouvelles tentatives qui a été défini. Elle déconnecte ensuite l'esclave.

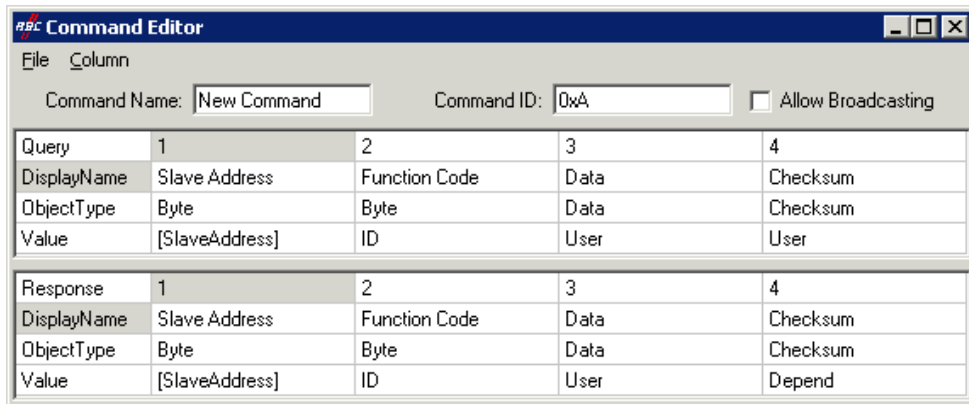
## 6. Configuration de la passerelle

### 6.12.3. Ajout d'une commande Modbus spéciale

En dehors des commandes Modbus standard dont il est question dans le chapitre précédent, il est possible de créer deux types de commandes Modbus spéciales : Des commandes Modbus utilisant le même modèle que les commandes standard, et des commandes Modbus dont la nature et le contenu des trames est entièrement modifiable par l'utilisateur.

#### 6.12.3.1. Commandes Modbus ayant pour modèle les commandes standard

Pour créer une commande de ce type, dans la fenêtre « Select Command » (voir chapitre 6.12.2), exécutez la commande « Add Command » du menu « Command ». La fenêtre reproduite en haut de la page suivante apparaît alors. Elle présente la structure des trames des requêtes et des réponses de la future commande, qui sera ensuite ajoutée à la liste des commandes Modbus disponibles. Cette structure comprend les éléments standard, c'est-à-dire les champs « Slave Address », « Function » et « Checksum », décrits dans les chapitres précédents.



The screenshot shows the 'ABC Command Editor' window. It has a menu bar with 'File' and 'Column'. Below the menu bar, there are two text input fields: 'Command Name: New Command' and 'Command ID: 0xA'. To the right of the 'Command ID' field is a checkbox labeled 'Allow Broadcasting' which is currently unchecked. Below these fields are two tables representing the structure of the query and response frames.

Query	1	2	3	4
DisplayName	Slave Address	Function Code	Data	Checksum
ObjectType	Byte	Byte	Data	Checksum
Value	[SlaveAddress]	ID	User	User

Response	1	2	3	4
DisplayName	Slave Address	Function Code	Data	Checksum
ObjectType	Byte	Byte	Data	Checksum
Value	[SlaveAddress]	ID	User	Depend

Reportez-vous au chapitre 2.12 « Command editor » du manuel d'utilisation de ABC-LUFP Config Tool, intitulé **AnyBus Communicator – User Manual**, pour de plus amples informations sur la création de commandes Modbus standard.

## 6. Configuration de la passerelle

---

### 6.12.3.2. Commandes Modbus personnalisables

Dans ABC-LUFP Config Tool, ces commandes sont appelées des « Transactions ». Contrairement aux exemples précédents, dans lesquels de nombreuses variables étaient fixes en raison de la commande Modbus sélectionnée, l'ensemble de la structure des trames de requêtes et de réponses associées à ces transactions repose sur les données présentes dans la mémoire de la passerelle. Ces champs de données présents dans la mémoire de la passerelle peuvent contenir des données constantes et comprises dans un intervalle, aux formats Byte, Word ou DWord et un champ final « Checksum ».

*(Reportez-vous au tableau des requêtes pour plus d'informations.)*

Toutes les données contenues dans les champs « Data » et « Variable Data » des requêtes et des réponses d'une commande de type « Transactions » sont gérées par le maître DeviceNet, y compris les champs « Slave address » et « Function » si ceux-ci sont placés dans un champ « Data ». Cela permet, par exemple, de gérer l'intégralité des champs des trames Modbus depuis le maître DeviceNet si l'ensemble des champs de la requête et de la réponse d'un élément « Transactions » (hors « Checksum ») sont des champs de type « Data » ou « Variable Data » pour les données de taille variable (ex. : la réponse à une requête utilisée pour lire un nombre variable de registres). Voir chapitre 6.12.3.3.

### AVERTISSEMENT

#### CHAMPS « DATA » MULTIPLES DANS UNE TRAME MODBUS

N'utilisez pas plus d'un champ « Data » par trame Modbus. Plusieurs champs « Data » utilisés dans une même trame Modbus risquent de ne pas être exécutés dans l'ordre approprié par la passerelle, provoquant des conséquences imprévues.

Il est préférable, pour le maître, de définir ces données comme un seul champ « Data », même si cela implique que les constantes intermédiaires soient intégrées au champ « Data » et ainsi échangées avec le maître.

Quant au champ « Variable Data », il ne peut y avoir qu'un seul champ de ce type dans une trame Modbus (requête ou réponse). Par conséquent, la commande « Add Variable Data » de ABC-LUFP Config Tool est désactivée si la trame actuelle inclut déjà un champ « Variable Data ».

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

Les constantes au format Byte, Word ou DWord placent les valeurs de ces constantes dans les trames des requêtes Modbus (constantes des éléments « Query ») ou les comparent à celles qui sont situées dans les réponses Modbus (constantes des éléments « Response »). Ces comparaisons servent à accepter (valeurs identiques) ou à refuser (valeurs différentes) les réponses Modbus de la même façon que dans le cas des commandes Modbus standard. Le maître DeviceNet n'a pas accès à ces constantes. Elles servent principalement à remplacer des champs tels que « Slave address », « Function », « Starting Address », etc.

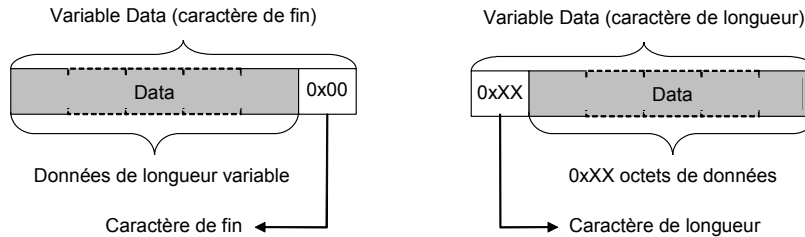
Reportez-vous à la section « Produce/Consume Menu » du chapitre **5.4.2 Transaction** et au chapitre **5.5 Frame objects** du manuel d'utilisation de ABC-LUFP Config Tool, intitulé **AnyBus Communicator – User Manual**, pour de plus amples informations sur la manipulation des commandes de type « Transactions ».

La configuration par défaut de la passerelle LUF9 comporte deux commandes de type « Transactions ». Il s'agit des commandes apériodiques de lecture et d'écriture de la valeur d'un paramètre d'esclave Modbus (forcément un départ-moteur TeSys U dans le cas de la configuration par défaut). Elles sont configurées pour le seul nœud « TeSys U n°1 », car l'adresse de l'esclave est pilotée par le maître DeviceNet via le premier octet du champ « Data », qui correspond au champ « Slave Address » des commandes Modbus standard. Cela permet au maître DeviceNet d'envoyer cette commande à tous les esclaves Modbus, en procédant esclave par esclave, par le biais du premier octet du champ « Data ». Le reste des trames de ces deux commandes est lui aussi placé dans le même champ « Data ». Le maître DeviceNet a donc accès à l'intégralité du contenu des trames de ces deux commandes.

## 6. Configuration de la passerelle

### 6.12.3.3. Utilisation de champs « Variable Data » dans des transactions

Un champ « Variable Data » est semblable à un champ « Data », mais n'a pas de longueur prédéfinie. Au lieu de cela, un caractère de longueur (ex. : un nombre d'octets) **ou** un caractère de fin indique la *longueur significative* du champ de données. Chaque champ « Variable Data » est également protégé par une propriété (« Maximum Data Length ») qui empêche tout dépassement en cas d'absence de caractère de fin à l'emplacement prévu ou lorsque la valeur du caractère de longueur est trop élevée.



Le caractère de fin ou de longueur d'un champ « Variable Data » inclus dans des requêtes de transactions doit être fourni par le maître DeviceNet, puisque ce dernier est le producteur de ces données.

Le caractère de fin ou de longueur d'un champ « Variable Data » inclus dans des réponses de transactions est généralement produit par la passerelle LUF7, et non par un esclave Modbus. Toutefois la réponse de la commande « Read Holding Registers » (commande Modbus 0x03) fait exception à la règle, puisque la valeur du champ « Byte count » correspondant peut servir de caractère de longueur (reportez-vous aux exemples à la fin de ce chapitre).

**REMARQUE :** Une requête ou une réponse de transaction ne peut comprendre qu'un seul champ « Variable Data ».

Le tableau ci-dessous décrit les propriétés des champs « Variable Data » :

Propriété	Remarques
Byte swap	Identique au champ « Data » standard. A titre de rappel, les trois valeurs possibles sont les suivantes : <ul style="list-style-type: none"> <li>• <b>No swapping</b> : Les données sont transmises dans le même ordre que celui de leur présence dans la mémoire de la passerelle.</li> <li>• <b>Swap 2 bytes</b> : Configuration par défaut d'un maître DeviceNet. Les octets à transmettre sont permutés deux à deux. <b>A utiliser par défaut dans le cas présent.</b></li> <li>• <b>Swap 4 bytes</b> : Les octets à transmettre sont permutés quatre à quatre.</li> </ul>
Data location	<p><b>Pour une requête</b> : Dans la mémoire des données de sortie de la passerelle (0x0202 à 0x02F3), adresse de début des données transmises par le maître DeviceNet à l'esclave Modbus. Ces données sont incluses directement dans la trame de la requête, à l'emplacement du champ actuel « Variable Data ».</p> <p><b>Pour une réponse</b> : Dans la mémoire des données d'entrée de la passerelle (0x0002 à 0x00F3), adresse de début des données transmises par l'esclave Modbus au maître DeviceNet. Ces données proviennent directement de la trame de la requête, à l'emplacement du champ actuel « Variable Data ».</p> <p><b>REMARQUE</b> : Dans les deux cas, le caractère de fin ou de longueur (s'il est utilisé) fait <i>partie des données</i>. Il peut ainsi figurer dans la mémoire des données d'entrée ou de sortie de la passerelle.</p>
End Character Value	Cette propriété est utilisée uniquement si « Object Delimiter » est défini sur « End Character » ou sur « End Character visible ». Elle permet de marquer la fin des données. Ce caractère spécifique est, bien évidemment, interdit à l'intérieur des données. Il est ainsi courant, par exemple, de terminer des chaînes de texte par un 0, car 0x00 ne peut pas être utilisé dans un texte écrit. On parle de représentation ASCII.

## 6. Configuration de la passerelle

	<i>Exemple</i> : la chaîne « ABC » devient { 0x41 , 0x42 , 0x43 , 0x00 } en ASCII.
Fill un-used Bytes	<p>Cette propriété est utilisée uniquement avec les champs « Variable Data » figurant dans les réponses de transactions, car les champs « Variable Data » inclus dans les requêtes sont mis à jour uniquement par le maître.</p> <p>Elle offre seulement deux options :</p> <ul style="list-style-type: none"> <li>• <b>Disabled</b> : Les données inutilisées (<i>c.-à-d.</i> les données situées après le dernier caractère ou au-delà du caractère de fin) ne sont pas mises à jour et conservent leur valeur actuelle.</li> <li>• <b>Enabled</b> : Les octets de données inutilisées contiennent la valeur définie dans « Filler Value ». Par exemple, si la valeur de la propriété « Filler Value » est 0xFF, toutes les données, situées après le dernier caractère ou au-delà du caractère de fin, sont alors définies sur 0xFF.</li> </ul>
Filler Value	Si la propriété « Fill un-used Bytes » est définie sur « Enabled » dans le champ « Variable Data » d'une réponse, cette valeur est alors copiée dans chaque octet situé après le dernier caractère ou au-delà du caractère de fin.
Maximum Data Length	<p>La combinaison des propriétés « Data location » et « Maximum Data length » détermine la mémoire d'entrée / de sortie utilisée pour l'échange de données entre le maître DeviceNet et l'esclave Modbus, exactement comme les propriétés « Data Location » et « Data length » des champs standard « Data ».</p> <p><b>REMARQUE</b> : La longueur maximale <i>doit</i> inclure le caractère de fin ou de longueur si l'un de ces deux caractères est utilisé (voir « Object Delimiter », ci-après). Si tel est le cas, ce caractère figure <i>toujours</i> dans la mémoire d'entrée / de sortie, même s'il n'est pas échangé avec l'esclave Modbus (<i>c.-à-d.</i> si l'<i>élément</i> « visible » optionnel n'a pas été choisi).</p>
Object Delimiter	<p>Cette propriété est essentielle, puisqu'elle détermine la méthode utilisée pour trier les données utiles parmi toutes les données d'entrée / de sortie associées au champ « Variable Data ». Elle offre cinq options :</p> <ul style="list-style-type: none"> <li>• <b>Length Character</b> : Le premier octet de la mémoire d'entrée / de sortie spécifie la longueur des données significatives (caractère de longueur exclus). Ce caractère <i>ne figure pas</i> dans la requête ou dans la réponse Modbus ; il est produit par la passerelle (en fonction de la longueur de la réponse Modbus) ou par le maître DeviceNet (qui, seul, met à jour les données de sortie).</li> <li>• <b>Length Character visible</b> : Identique à l'option « Length Character » à la différence que ce caractère est intégré à la requête ou à la réponse Modbus ; il est produit par l'esclave Modbus (dans la réponse) ou par le maître DeviceNet (dans la requête).</li> <li>• <b>End Character</b> : Les données significatives se terminent à la première occurrence de la valeur de la propriété « End Character Value ». Ce caractère <i>ne figure pas</i> dans la requête ou dans la réponse Modbus ; il est produit par la passerelle (en fonction de la longueur de la réponse Modbus) ou par le maître DeviceNet (qui, seul, met à jour les données de sortie).</li> <li>• <b>End Character visible</b> : Identique à l'option « End Character » à la différence que ce caractère est intégré à la requête ou à la réponse Modbus ; il est produit par l'esclave Modbus (dans la réponse) ou par le maître DeviceNet (dans la requête).</li> <li>• <b>No Character</b> : Cette option est exclusivement réservée aux réponses. En cas de réception d'une réponse contenant des données variables, la passerelle copie simplement les données depuis la trame vers sa mémoire d'entrée. Par conséquent, le maître DeviceNet ne peut pas déterminer la véritable longueur des données significatives (<i>c.-à-d.</i> les données qui ont été mises à jour).</li> </ul>

## 6. Configuration de la passerelle

---

### Exemple 1 :

Configuration des communications entre une passerelle LUPF9 et un seul esclave Modbus (un départ-moteur TeSys U situé à l'adresse 1 sur le sous-réseau Modbus et intitulé « TeSys U n°1 ») :

- Les deux premiers octets de la mémoire d'entrée (0x0000-0x0001) et de la mémoire de sortie (0x0200-0x0201) de la passerelle sont réservés à l'initialisation et aux diagnostics (voir chapitre 5), en mode « Diagnostic et commande » (« Control/Status Word = Enabled but no startup lock » pour l'élément « ABC-LUPF »).
- 1 commande « Read Holding Registers » (FC 0x03) : Commande périodique (« Update mode = Cyclically » et « Update time (10ms) = 30 » pour la requête) permettant d'obtenir l'état du départ-moteur TeSys U (« Starting register address = 0x01C7 = 455 » et « Number of registers = 0x0001 » dans la requête ; « Byte count = 0x02 » dans la réponse) ; la valeur de cet état est transmise aux adresses 0x0002-0x0003 de la mémoire d'entrée de la passerelle (« Data length = 0x0002 » et « Data location = 0x0002 » pour le champ « Data » de la réponse « Response »).
- 1 commande « Preset Multiple Regs » (FC 0x10) : Commande périodique (« Update mode = Cyclically » et « Update time (10ms) = 30 » pour la requête) permettant de définir la commande du départ-moteur TeSys U (« Starting register address = 0x02C0 = 704 » « Number of registers = 0x0001 » et « Byte count = 0x02 » dans la requête ; « Starting register address = 0x02C0 = 704 » et « Number of registers = 0x0001 » dans la réponse) ; la valeur de cette commande est transmise aux adresses 0x0202-0x0203 de la mémoire de sortie de la passerelle (« Data length = 0x0002 » et « Data location = 0x0202 » pour le champ « Data » de la requête « Query »).
- 1 commande « Transactions » : Commande périodique (« Update mode = Cyclically » et « Update time (10ms) = 100 » pour la requête) permettant d'obtenir de 1 à 5 registres d'état (nombre exact dans 0x0204-0x0205) du départ-moteur TeSys U (début au registre 455 / 0x01C7) ; la valeur de ces registres est transmise aux adresses 0x0006-0x000F de la mémoire d'entrée de la passerelle (longueur de 2, 4, 6, 8 ou 10 octets selon le nombre de registres réellement lus, pour 10 octets maximum). Le contenu de cette commande est détaillé ci-après afin de faciliter la compréhension de l'exemple qui s'y rapporte :
  - La requête est constituée des champs suivants, dans cet ordre :
    - 1 champ « Byte, Constant » renommé « Address » : 0x01 (adresse de l'esclave Modbus).
    - 1 champ « Byte, Constant » renommé « Function code » : 0x03 (code fonction d'une commande « Read Holding Registers »).
    - 1 champ « Word, Constant » renommé « Register Address » : 0x01C7 (pour émuler le champ « Starting register address » de FC 0x03).
    - 1 champ « Data » où « Data length = 0x0002 » et « Data location = 0x0204 » (pour remplacer le champ « Number of registers » de FC 0x03) ; le maître DeviceNet utilise ce champ de données de sortie pour définir le nombre de registres d'état (de 1 à 5) à lire provenant de l'esclave TeSys U.
    - 1 champ « Checksum » (obligatoire : CRC à 0x0000).
  - La réponse est constituée des champs suivants, dans cet ordre :
    - 1 champ « Byte, Constant » renommé « Address » : 0x01 (adresse de l'esclave Modbus).
    - 1 champ « Byte, Constant » renommé « Function code » : 0x03 (code fonction d'une commande « Read Holding Registers »).
    - 1 champ « Byte, Limits » renommé « Byte count », où « Minimum Value = 0x02 » et « Maximum Value = 0x0A » (pour émuler le champ « Byte count » de FC 0x03) ; ces valeurs limitent la lecture de réponse de 1 à 5 registres (2 à 10 octets).
    - 1 champ « Variable Data » en remplacement du champ standard « Data » généralement utilisé pour FC 0x03 ; ses propriétés sont définies comme suit :
      - « Byte swap = Swap 2 bytes »..... Cas par défaut d'un maître DeviceNet.
      - « Data location = 0x0005 » ..... Les données commencent à 0x0005 avec un caractère de longueur (voir ci-après) ; ainsi, *les données significatives commencent réellement à 0x0006* (les données 16 bits sont donc alignées sur des adresses mémoire paires).
      - « End Character Value = 0x00 » ..... Non utilisé dans le cas présent.

## 6. Configuration de la passerelle

- « Fill un-used Bytes = Enabled »..... Dans cet exemple, les données d'entrée non mises à jour, lues à partir de l'esclave TeSys U, sont définies sur 0xFF (« Filler Value »).
- « Filler Value = 0xFF » ..... La valeur copiée dans les données non mises à jour provenant de la trame de la réponse, *c.-à-d.* les données situées au-delà du dernier caractère comme l'indique l'option « Length Character ».
- « Maximum Data length = 0x000B » 11 octets maximum doivent être acceptés et affectés à la mémoire d'entrée (de 0x0005 à 0x000F) ; le premier octet correspond au caractère de longueur et les 10 autres aux données significatives provenant de la trame de la réponse envoyée par l'esclave Modbus.
- « Object Delimiter = Length Character » Ce mode établit que le premier octet de données d'entrée (0x0005, dans le cas présent) correspond à la longueur des données significatives (0x0005 exclus) et qu'en tant que caractère non « visible », cet octet ne figure pas dans la trame de la réponse, mais est évalué par la passerelle en fonction de la *longueur réelle* de la trame de la réponse.
  - 1 champ « Checksum » (obligatoire : CRC à 0x0000).

Dans cette configuration, le contenu de la mémoire de la passerelle est le suivant :

Mémoire d'entrée (16 octets)	
0x0000-0x0001	Passerelle : Mot d'état
0x0002-0x0003	TeSys U : Registre d'état (455)
0x0004	Rechange / Non utilisé
0x0005	Longueur données significatives
0x0006-0x0007	1er registre d'état (455)
0x0008-0x0009	2ème registre d'état (456)
0x000A-0x000B	3ème registre d'état (457)
0x000C-0x000D	4ème registre d'état (458)
0x000E-0x000F	5ème registre d'état (459)

Mémoire de sortie (6 octets)	
0x0200-0x0201	Passerelle : Mot de commande
0x0202-0x0203	TeSys U : Registre de commande (704)
0x0204-0x0205	Nombre de registres à lire (1 à 5).

Utilisez l'outil de configuration DeviceNet (ex. : RSNetWorx) pour redimensionner les données d'entrée / de sortie échangées entre le maître (ex. : 1747-SDN Scanner Module) et la passerelle LUF9 ; définissez le champ « Rx Size: » (entrée) de la connexion « Polled » sur 16 octets et le champ « Tx Size: » (sortie) de la connexion « Polled » sur 6 octets.

Sous RSNetWorx et RSLogix 500, si une carte du 1747-SDN Scanner Module est insérée dans un automate SLC500 **Allen Bradley**, ces entrées / sorties se présentent comme suit :

Entrées (8 mots)	
I:1.1	Passerelle : Mot d'état
I:1.2	TeSys U : Registre d'état (455)
I:1.3	Longueur de données significatives (bits 0-7)
I:1.4	1er registre d'état (455)
I:1.5	2ème registre d'état (456)
I:1.6	3ème registre d'état (457)
I:1.7	4ème registre d'état (458)
I:1.8	5ème registre d'état (459)

Sorties (3 mots)	
O:1.1	Passerelle : Mot de commande
O:1.2	TeSys U : Registre de commande (704)
O:1.3	Nombre de registres à lire (1 à 5).

## 6. Configuration de la passerelle

---

Pour un départ-moteur commandé en mode RUN (O:1.2 = 0x0001), il est possible de lire l'état correspondant dans I:1.2 (0x0043), mais également depuis I:1.4 à I:1.8, selon le nombre de registres réellement lus (O:1.3 = 0x0001 à 0x0005) :

Entrées résultantes	Valeur de O:1.3				
	0x0001	0x0002	0x0003	0x0004	0x0005
I:1.3	0x●●02	0x●●04	0x●●06	0x●●08	0x●●0A
I:1.4	0x0043	0x0043	0x0043	0x0043	0x0043
I:1.5	0xFFFF	0x0000	0x0000	0x0000	0x0000
I:1.6	0xFFFF	0xFFFF	0x000D	0x000D	0x000D
I:1.7	0xFFFF	0xFFFF	0xFFFF	0x0001	0x0001
I:1.8	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0x0000

Notez que la passerelle définit tous les octets situés au-delà du dernier octet significatif sur 0xFF (« Filler Value »).

### Exemple 2 :

La configuration décrite dans l'*exemple 1* est la même pour ce deuxième exemple, mises à part les deux exceptions suivantes :

- Dans le champ « Variable Data », « Byte, Limits » est renommé « Byte count », « Minimum Value = 0x02 » et « Maximum Value = 0x0A » ; ce champ est supprimé de la réponse, car il est à présent inclus dans les données provenant de la trame de la réponse et copié dans la mémoire d'entrée de la passerelle (reportez-vous aux valeurs de I:1.3 / 0x0005 pour vous en assurer).
- Dans le champ « Variable Data », « Object Delimiter = Length Character » devient « Object Delimiter = Length Character visible » ; la passerelle récupère ainsi le caractère de longueur (1 octet) de la trame de la réponse de l'esclave Modbus au lieu de l'évaluer avec la longueur restante de la trame de la réponse.

Etant donné que ces deux modifications se compensent mutuellement dans le cas spécifique d'une commande « Read Holding Register », les résultats décrits à la fin de l'*exemple 1* s'appliquent également ici.



## 6. Configuration de la passerelle

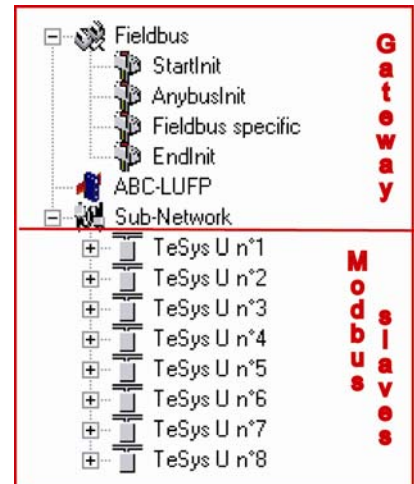
### 6.13. Configuration des caractéristiques générales de la passerelle

Cette opération concerne les caractéristiques générales de la passerelle (éléments « Fieldbus » à « Sub-Network »), alors que les chapitres précédents s'attachaient à décrire la configuration des esclaves Modbus (éléments situés sous l'élément « Sub-Network »).

L'élément « Fieldbus » décrit le réseau amont, c'est-à-dire le réseau DeviceNet dans le cas de la passerelle LUF9.

Les éléments « ABC-LUFP » et « Sub-Network » décrivent le réseau aval, c'est-à-dire le réseau Modbus RTU dans le cas de la passerelle LUF9, et permettent d'identifier la version du logiciel présent dans la passerelle.

La configuration de ces trois éléments, ainsi que les commandes auxquelles ils donnent accès, sont décrites dans les trois chapitres suivants.

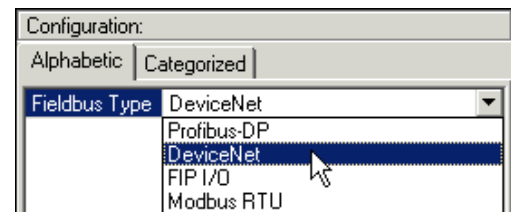


#### 6.13.1. Élément « Fieldbus »

Sous cet élément figure la liste des télégrammes (appelés « mailboxes ») configurés par défaut. Ces éléments ne sont pas décrits ici, car ils sont propres à la gestion interne de la passerelle. Ces « mailboxes » configurés par défaut ne peuvent être ni modifiés, ni supprimés. Leur nombre et leur nature dépendent du type du réseau amont.

Lorsque l'élément « Fieldbus » est sélectionné, vous avez la possibilité de sélectionner le type du réseau amont : « DeviceNet » avec la passerelle LUF9.

Si le réseau sélectionné ne correspond pas à la passerelle, un message d'erreur s'affiche et la configuration n'est pas chargée.

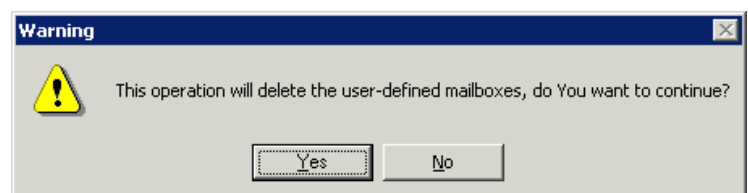


Si votre PC est relié à la passerelle à l'aide du câble PowerSuite et que vous utilisez ABC-LUFP Config Tool en mode « connecté » dès le démarrage de ABC-LUFP Config Tool, le type du réseau amont sera automatiquement détecté.

L'unique commande accessible depuis le menu « Fieldbus » est la commande « Restore Default Mailboxes ». Il est recommandé d'utiliser cette commande en cas d'insertion accidentelle d'un « mailbox » défini par l'utilisateur avec l'élément « Fieldbus ». Etant donné que l'utilisation de « mailboxes » définis par l'utilisateur n'est pas décrite dans le présent document, seuls les « mailboxes » par défaut de DeviceNet doivent être définis avec l'élément « Fieldbus », dans l'ordre suivant :

- StartInit
- AnybusInit
- Fieldbus specific
- EndInit

Si un autre « mailbox » figure également dans la liste, exécutez la commande « Restore Default Mailboxes ». Confirmez ensuite l'opération en cliquant sur le bouton « Yes » de la fenêtre de confirmation / d'avertissement affichée.

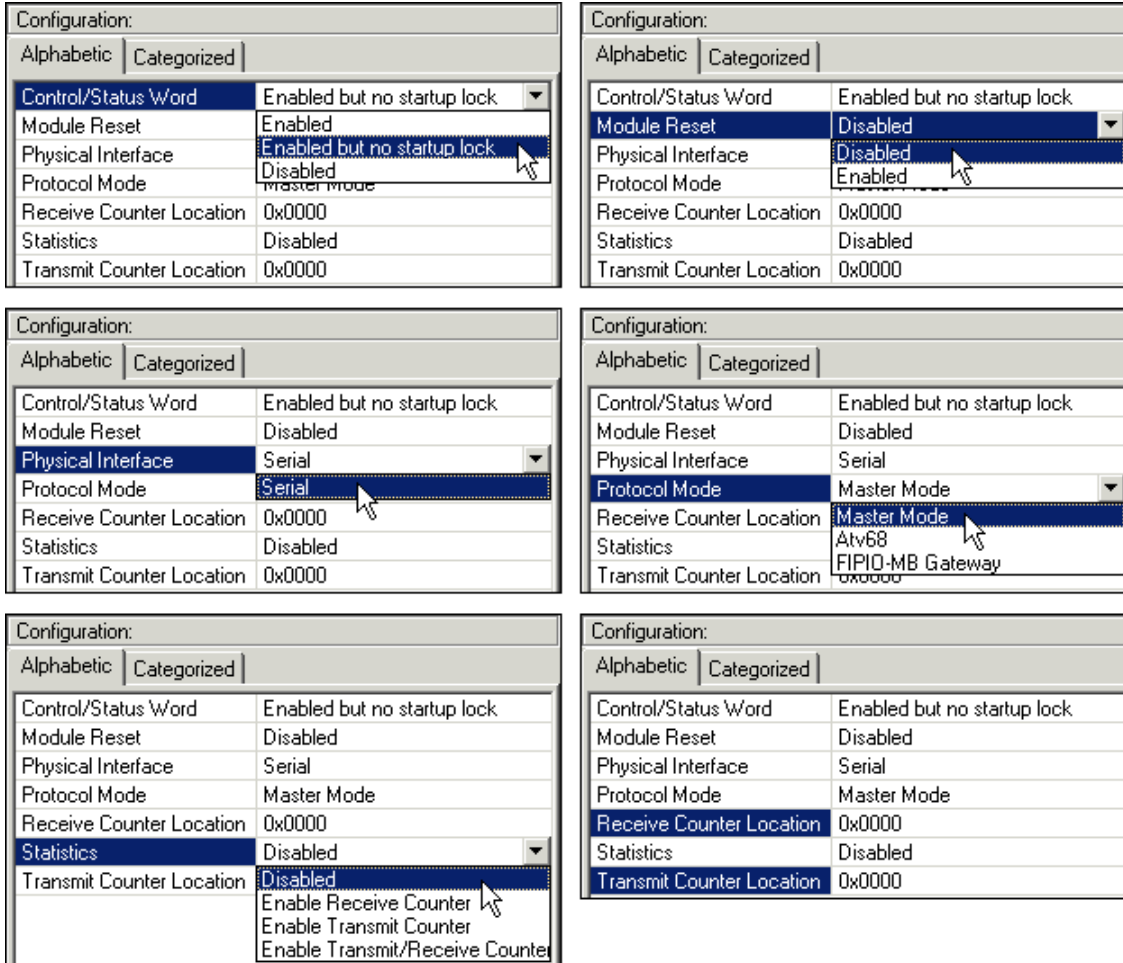


# 6. Configuration de la passerelle

## 6.13.2. Élément « ABC-LUFP »

La seule commande accessible depuis le menu « ABC-LUFP » est la commande « Disconnect » (ou « Connect » si vous êtes en mode « déconnecté ») ; reportez-vous au chapitre 6.3 Connexion / déconnexion de la passerelle pour en savoir plus sur les modes « connecté » et « déconnecté ».

Dans la configuration de l'élément « ABC-LUFP » de la passerelle LUFP9 , les propriétés « Physical Interface » et « Protocol Mode » *ne doivent pas* être modifiées. Elles doivent toujours être définies, respectivement, sur « Serial » et « Master Mode ».



## 6. Configuration de la passerelle

---

Les sept propriétés suivantes permettent de configurer certains aspects système de la passerelle :

- Control/Status Word : Les trois possibilités disponibles pour cette propriété sont décrites dans le chapitre 5.
- Module Reset : Par défaut, cette propriété empêche la passerelle de se réinitialiser lorsqu'un problème de fonctionnement interne se produit. La modification de cette option est principalement destinée à un usage de type « laboratoire ».
- Physical Interface : L'unique possibilité offerte pour cette propriété indique que l'interface physique du réseau aval de la passerelle (Modbus) est une liaison série.
- Protocol Mode : Cette propriété ne doit pas être modifiée, car elle indique le type de protocole utilisé sur le réseau aval de la passerelle. Dans le cas de la passerelle LUFP9, « Master Mode » doit impérativement être sélectionnée. Les autres possibilités offertes sont réservées à d'autres produits de la même famille que cette passerelle.
- Statistics : Cette propriété détermine la présence ou l'absence des deux compteurs de réception et de transmission dans la mémoire d'entrée de la passerelle (voir ci-après). Elle offre les quatre possibilités suivantes :
  - Disabled : Les deux propriétés « Receive Counter Location » et « Transmit Counter Location » sont ignorées.
  - Enable Receive Counter : Seule la propriété « Receive Counter Location » est utilisée par la passerelle.
  - Enable Transmit Counter : Seule la propriété « Transmit Counter Location » est utilisée par la passerelle.
  - Enable Transmit/Receive Counter : Les deux propriétés « Receive Counter Location » et « Transmit Counter Location » sont utilisées par la passerelle.
- Receive Counter Location : Cette propriété est utilisée uniquement par la passerelle si « Statistics = Enable Receive Counter » ou « Statistics = Enable Transmit/Receive Counter ». Elle représente l'adresse mémoire d'entrée 1 octet (de 0x0000 à 0x00F3) où est copié le compteur de réponses Modbus. Comme tout autre donnée de mémoire d'entrée utilisée, cet octet augmente le flux de données échangées avec le maître DeviceNet. Il s'agit d'un compteur modulo 256 (c.-à-d. qu'il redémarre à 0 lorsqu'il dépasse 255) qui est mis à jour chaque fois que la passerelle reçoit une trame Modbus.
- Transmit Counter Location : Cette propriété est utilisée uniquement par la passerelle si « Statistics = Enable Transmit Counter » ou « Statistics = Enable Transmit/Receive Counter ». Elle représente l'adresse mémoire d'entrée 1 octet (de 0x0000 à 0x00F3) où est copié le compteur de requêtes Modbus. Comme tout autre donnée de mémoire d'entrée utilisée, cet octet augmente le flux de données échangées avec le maître DeviceNet. Il s'agit d'un compteur modulo 256 (c.-à-d. qu'il redémarre à 0 lorsqu'il dépasse 255) qui est mis à jour chaque fois que la passerelle transmet une trame Modbus, nouvelles tentatives incluses.

Enfin, le menu « Help » contient une commande utile permettant de vérifier la version logicielle de la passerelle LUFP9 (l'élément « ABC-LUFP »), uniquement en mode « connecté », mais aussi d'afficher la version de ABC-LUFP Config Tool.

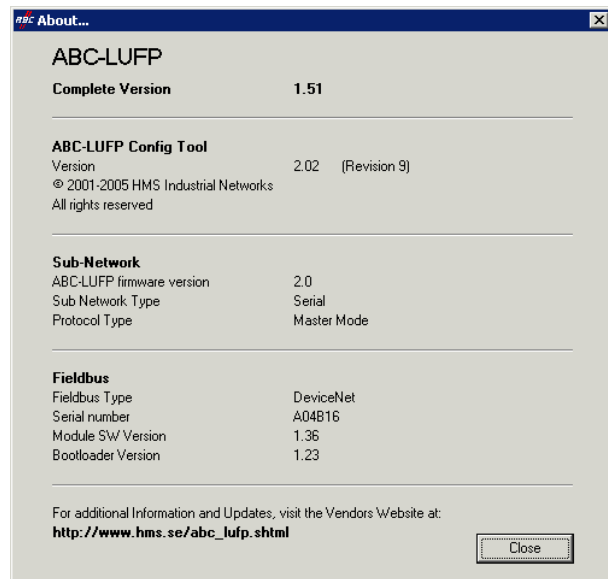
## 6. Configuration de la passerelle

Pour obtenir ces informations, exécutez la commande « About... » dans le menu « Help ». L'exemple ci-contre est en mode « connecté ».

En mode « déconnecté », toutes les versions et les informations des catégories « Sub-Network » et « Fieldbus » sont remplacées par la mention « Unknown », car elles ne sont pas disponibles avec une passerelle connectée existante.

Le texte [http://www.hms.se/abc\\_lufp.shtml](http://www.hms.se/abc_lufp.shtml) est un lien hypertexte. Si vous cliquez dessus, vous êtes directement dirigé vers la page Web **Schneider Electric** consacrée aux passerelles ABC-LUFP.

Cette page contient de nombreux éléments téléchargeables relatifs à la famille de passerelles LUFP, ainsi que la dernière version de ABC-LUFP Config Tool.

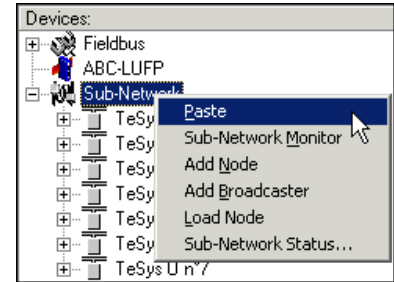


## 6. Configuration de la passerelle

### 6.13.3. Élément « Sub-Network »

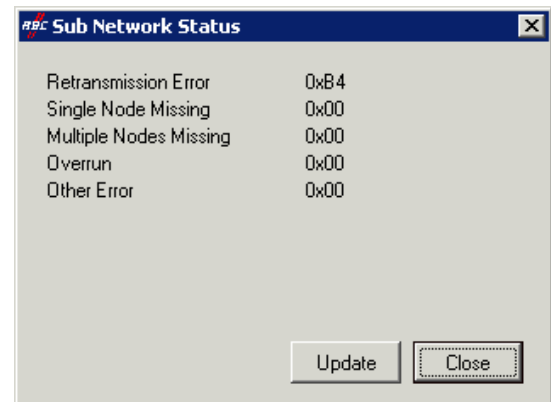
Les cinq commandes disponibles dans le menu « Sub-Network » sont les suivantes :

- « Paste » : Ajoute une copie du dernier nœud copié (après avoir exécuté la commande « Copy » sur un nœud existant) ou un réplica du nœud coupé (après avoir exécuté la commande « Cut ») à la liste des nœuds de l'élément « Sub-Network ». Cette commande est disponible uniquement si un nœud a préalablement été copié ou coupé et si la limite de 8 nœuds n'a pas déjà été atteinte.
- « Sub-Network Monitor » : Permet de consulter la correspondance entre les données des commandes Modbus et le contenu de la mémoire de la passerelle. Des exemples d'utilisation de cette commande sont présentés dans les chapitres 6.9.3, 6.9.4 et 6.10.
- « Add Node » : Permet d'ajouter un nouveau nœud sur le réseau aval Modbus. Chaque nœud correspond à un esclave Modbus différent. Cette commande n'est pas disponible s'il y a déjà 8 esclaves Modbus, ce qui est le cas dans la configuration par défaut de la passerelle.
- « Add Broadcaster » : Permet d'ajouter un nœud de diffusion (voir chapitre 6.14).
- « Load Node » : Permet d'ajouter un nœud pré-configuré sur le réseau aval Modbus. La configuration de ce nœud est contenue dans un fichier XML (voir section « Importation/exportation de la configuration d'un esclave Modbus » du chapitre 6.8). Cette commande n'est pas disponible s'il y a déjà 8 esclaves Modbus, ce qui est le cas dans la configuration par défaut de la passerelle.



- « Sub-Network Status... » : En mode « connecté » (voir chapitre 6.13.2), cette commande permet d'afficher une fenêtre récapitulant les valeurs des compteurs d'erreurs de la passerelle. Ces compteurs sont également utilisés par la passerelle pour mettre à jour la valeur de son mot d'état (voir chapitre 5.5). Le bouton « Update » permet d'actualiser les valeurs de ces compteurs.

Lorsque cette commande est exécutée en mode « déconnecté », toutes les valeurs affichées sont remplacées par la mention « Unknown » pour signifier qu'elles ne peuvent pas être lues sur la passerelle. Le bouton « Update » devient alors inaccessible.



**REMARQUE** : La fenêtre « Sub Network Status » peut être utile pour détecter des problèmes sur le sous-réseau Modbus. Par conséquent, si le nombre d'erreurs de retransmission augmente lorsque vous cliquez sur le bouton « Update », cela indique l'absence d'un ou plusieurs esclaves, des problèmes de vitesse et de raccordement Modbus ou l'indisponibilité de commandes et/ou de transactions. Etant donné que les erreurs de retransmission tendent à réduire les performances générales des communications Modbus, vous devez prendre les mesures nécessaires pour empêcher l'augmentation de ce type d'erreurs.

## 6. Configuration de la passerelle

Lorsque l'élément « Sub-Network » est sélectionné, vous avez accès à l'ensemble des options permettant de paramétrer le format du protocole de communication de la passerelle sur le réseau Modbus. Les différents paramétrages que vous pouvez effectuer sont décrits ci-dessous. L'ensemble des esclaves Modbus présents doivent supporter ce paramétrage et être configurés de manière appropriée.

- Bitrate (bits/s) : La passerelle supporte un nombre limité de vitesses de communication. Sélectionnez celle qui convient à l'esclave le plus lent.
- Data bits : 8 bits (obligatoire).
- Parity : Choisissez la parité en fonction du format retenu pour les communications sur votre réseau Modbus.
- Physical standard : RS485 (obligatoire).
- Stop bits : 1 bit (parité paire ou impaire) ou 2 bits (sans parité).

Configuration:	
Alphabetic	Categorized
Bitrate (bits/s)	19200
Data bits	1200
Parity	2400
Physical standard	4800
Stop bits	9600
	19200

Configuration:	
Alphabetic	Categorized
Bitrate (bits/s)	19200
Data bits	8
Parity	7
Physical standard	8
Stop bits	1

Configuration:	
Alphabetic	Categorized
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	None
Stop bits	Odd
	Even

Configuration:	
Alphabetic	Categorized
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	RS485
Stop bits	RS232
	RS485

Configuration:	
Alphabetic	Categorized
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	RS485
Stop bits	1
	2

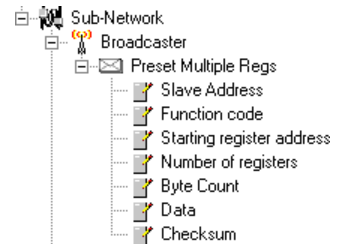
## 6. Configuration de la passerelle

---

### 6.14. Ajout d'un nœud de diffusion

Un nœud de diffusion ne correspond à aucun esclave Modbus en particulier, car il s'applique à **tous** les esclaves Modbus. Toutes les commandes qui seront configurées pour ce nœud seront émises avec le champ « Slave Address » égal à 0x00. Cela signifie que tous les esclaves exécuteront la commande, mais qu'aucun d'entre eux n'y répondra.

Pour ajouter un nœud de diffusion, sélectionnez l'élément « Sub-Network », puis exécutez la commande « Add Broadcaster » du menu « Sub-Network ». Le nœud de diffusion ainsi créé ne compte pas dans la limite du nombre de nœuds configurables. Un exemple simple figure ci-contre :




L'ajout et le paramétrage d'une commande Modbus dans la liste des commandes du nœud de diffusion sont effectués de la même manière que pour les autres nœuds, aux différences suivantes près :

- La liste des commandes Modbus standard qu'il est possible d'utiliser en diffusion est réduite. Seules les fonctions 0x06 et 0x10 peuvent être utilisées (voir la liste du chapitre 6.12.2).
- La commande est constituée d'une requête, mais ne comporte aucune réponse. La requête porte le nom de la commande elle-même, au lieu de l'appellation « Query ». De plus, chaque commande de diffusion ne consomme qu'une seule des 100 requêtes et réponses admises par la passerelle, puisqu'il n'y a aucune réponse possible pour une telle commande.
- La valeur du champ « Slave Address » de la trame de la requête est égale à 0x00.


Reportez-vous au chapitre 6.12.2.2, pour plus d'informations sur la configuration d'une requête Modbus.

# Annexe A : Caractéristiques techniques

## Environnement

Dimensions (hors connecteurs)	Hauteur : 120 mm      Largeur : 27 mm      Profondeur : 75 mm
Apparence externe	Boîtier plastique avec dispositif de fixation à un rail DIN.
Couple de serrage	Connecteur d'alimentation : compris entre 0,56 et 0,79 N-m.
Alimentation	 24 V régulé à $\pm 10\%$ Consommation maximale : 280 mA (généralement autour de 100 mA)
Classe de protection	IP20
Humidité relative maximale	95% sans condensation ni ruissellement, conformément à la norme IEC 68-2-30
Température de l'air ambiant autour de l'appareil, en milieu sec	Conformément aux normes IEC 68-2-1 Ab, IEC 68-2-2 Bb et IEC 68-2-14 Nb : <ul style="list-style-type: none"> <li>• Stockage :            <math>-55\text{ °C} (\pm 3)</math>            à            <math>+85\text{ °C} (\pm 2)</math></li> <li>• Fonctionnement : <math>-5\text{ °C} (\pm 3)</math>            à            <math>+55\text{ °C} (\pm 2)</math></li> </ul> <p style="text-align: center;">...</p>
UL	Certificat E 214107 Catégorie « type ouvert » Le produit doit être installé dans une armoire électrique ou dans un endroit équivalent.
CE	Certifié conforme aux normes européennes, sauf avis contraire.
Compatibilité électromagnétique (CEM) : Emission	Conforme à la norme EN 50 081-2:1993 (environnement industriel) Testé selon la classe A en rayonnement de la norme EN 55011:1990
Compatibilité électromagnétique (CEM) : Immunité	Conforme aux normes EN 50 082-2:1995 et EN 61 000-6-2:1999 (environnement industriel) Testé selon les normes EN 50 204:1995, EN 61000-4-2:1995, EN 61000-4-3:1996, EN 61000-4-4:1995, EN 61000-4-5:1995 et EN 61000-4-6:1996.

## Caractéristiques de communication

Réseau « amont »	DeviceNet
Réseau « aval »	Modbus RTU
Caractéristiques DeviceNet	<ul style="list-style-type: none"> <li>• Topologie du réseau : Topologie linéaire multipoints (bus) avec terminaisons de ligne adaptées (impédance de <math>121\ \Omega \pm 1\% \frac{1}{4} W</math>).</li> <li>• Média physique : Quatre types de câbles DeviceNet spécifiques, avec alimentation  24 V intégrée : <ul style="list-style-type: none"> <li>① Câble cylindrique épais à double paire torsadée      ③ Câble plat</li> <li>② Câble cylindrique fin à double paire torsadée      ④ Câble de type « KwikLink »</li> </ul> </li> <li>• Vitesse de communication : 125, 250 ou 500 kbits/s</li> <li>• Longueur totale maximale du réseau : <ul style="list-style-type: none"> <li>500 m à 125 kbits/s</li> <li>250 m à 250 kbits/s</li> <li>100 m à 500 kbits/s</li> </ul> </li> <li>• Nombre maximum d'abonnés : 64</li> <li>• Transactions : Jusqu'à 8 octets de données par trame.</li> <li>• Possibilité de connecter ou de déconnecter un abonné sans affecter les communications entre les autres abonnés.</li> </ul>



# Annexe A : Caractéristiques techniques

Spécificités DeviceNet de la passerelle LUF9	<ul style="list-style-type: none"> <li>• La passerelle LUF9 est un abonné DeviceNet de type « group two only server » (reportez-vous à <i>DeviceNet Specifications</i>).</li> <li>• Support de la fragmentation pour les transactions exigeant plus de 8 octets de données.</li> <li>• Connexions supportées :             <ul style="list-style-type: none"> <li>1 connexion « Explicit Connection »</li> <li>1 connexion « Polled Command/Response »</li> <li>1 connexion « Bit Strobed Command/Response »</li> <li>1 connexion « Change-of-State / Cyclic »</li> </ul> </li> <li>• Vitesse de communication configurée à l'aide de 2 commutateurs.</li> <li>• Adresse DeviceNet (MAC ID) de la passerelle configurée à l'aide de 6 commutateurs (adresse comprise entre 0 et 63).</li> <li>• Configuration facilitée par l'usage d'un fichier EDS spécifique.</li> </ul>
Caractéristiques Modbus RTU	<ul style="list-style-type: none"> <li>• Média physique : Liaison série RS485</li> <li>• Topologie du réseau : Topologie linéaire multipoints avec terminaisons de ligne adaptées (impédance de 120 <math>\Omega</math> en parallèle avec une capacité de 1 nF)</li> <li>• Vitesse de communication : 1 200 à 57 600 bits/s</li> <li>• Bits de données : 8</li> <li>• Adresses des abonnés : 1 à 247. Adresse 0 réservée à la diffusion. Adresses 65, 126 et 127 réservées si des produits de la gamme Variation de Vitesse de <i>Schneider Electric</i> sont utilisés sur le même réseau Modbus.</li> <li>• Temps de silence : Equivalent à la transmission de 3,5 caractères.</li> </ul>

## AVERTISSEMENT

### UTILISATION D'ADRESSES MODBUS RESERVEES

N'utilisez pas les adresses Modbus 65, 126 ou 127 si les esclaves Modbus d'une passerelle comportent un système de variation de vitesse Schneider Electric, tel qu'un démarreur Altistart ou un variateur Altivar. Les périphériques Altistart et Altivar réservent ces adresses pour d'autres communications et l'utilisation de ces adresses dans un tel système peut avoir des conséquences imprévues.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

Spécificités Modbus RTU de la passerelle LUF9	<ul style="list-style-type: none"> <li>• Nombre maximum d'abonnés (hors passerelle) : 8 esclaves Modbus.</li> <li>• Nombre maximum de commandes configurées : Jusqu'à 100 requêtes et réponses Modbus configurées pour la même passerelle à l'aide de ABC-LUF9 Config Tool.</li> <li>• Vitesse de communication : 1 200, 2 400, 4 800, 9 600 ou 19 200 bits/s ; configurée à l'aide de ABC-LUF9 Config Tool.</li> <li>• Temps de silence : Impossibilité d'augmenter le temps de silence de la passerelle.</li> <li>• Parité : Aucune, paire ou impaire, configurée à l'aide de ABC-LUF9 Config Tool.</li> <li>• Bits de start : 1 bit uniquement.</li> <li>• Bits de stop : 1 ou 2 bits, configuration à l'aide de ABC-LUF9 Config Tool.</li> </ul>												
Structure de la mémoire de la passerelle LUF9 :  <p style="text-align: center;"><b>Entrées</b></p>	<ul style="list-style-type: none"> <li>• 2 octets pour le diagnostic des erreurs du réseau aval par la passerelle (voir chapitre 5).</li> <li>• 510 octets accessibles par le maître DeviceNet sous la forme de données d'entrée (voir Annexe B : Configuration par défaut, Zone mémoire des données d'entrée, pour l'utilisation par défaut de ces données d'entrée).</li> </ul> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="text-align: center;">Adresses</th> <th style="text-align: center;">Zone des données d'Entrée</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0x0000</td> <td style="text-align: center;">Mot d'état de la passerelle</td> </tr> <tr> <td style="text-align: center;">0x0001</td> <td style="text-align: center;">(sauf si « Control/Status Byte » = « Disabled »)</td> </tr> <tr> <td style="text-align: center;">0x0002</td> <td style="text-align: center;">Entrées accessibles par le maître DeviceNet</td> </tr> <tr> <td style="text-align: center;">:</td> <td style="text-align: center;">510 octets</td> </tr> <tr> <td style="text-align: center;">0x01FF</td> <td style="text-align: center;">1 zone de données d'entrée</td> </tr> </tbody> </table>	Adresses	Zone des données d'Entrée	0x0000	Mot d'état de la passerelle	0x0001	(sauf si « Control/Status Byte » = « Disabled »)	0x0002	Entrées accessibles par le maître DeviceNet	:	510 octets	0x01FF	1 zone de données d'entrée
Adresses	Zone des données d'Entrée												
0x0000	Mot d'état de la passerelle												
0x0001	(sauf si « Control/Status Byte » = « Disabled »)												
0x0002	Entrées accessibles par le maître DeviceNet												
:	510 octets												
0x01FF	1 zone de données d'entrée												

# Annexe A : Caractéristiques techniques

<p>Structure de la mémoire de la passerelle LUF9 :</p> <p><b>Sorties</b></p>	<ul style="list-style-type: none"> <li>• 2 octets pour l'activation ou l'inhibition du réseau aval par la passerelle (voir chapitre 5).</li> <li>• 510 octets accessibles par le maître DeviceNet sous la forme de données de sortie (voir Annexe B : Configuration par défaut, Zone mémoire des données de sortie, pour l'utilisation par défaut de ces données de sortie).</li> </ul> <table border="1" data-bbox="529 415 1365 625"> <thead> <tr> <th>Adresses</th> <th>Zone des données de Sortie</th> </tr> </thead> <tbody> <tr> <td>0x0200</td> <td rowspan="2">Mot de commande du maître DeviceNet (sauf si « Control/Status Byte » = « Disabled »)</td> </tr> <tr> <td>0x0201</td> </tr> <tr> <td>0x0202</td> <td rowspan="4">Sorties accessibles par le maître DeviceNet 510 octets 1 zone de données de sortie</td> </tr> <tr> <td>:</td> </tr> <tr> <td>0x03FF</td> </tr> <tr> <td></td> </tr> </tbody> </table>	Adresses	Zone des données de Sortie	0x0200	Mot de commande du maître DeviceNet (sauf si « Control/Status Byte » = « Disabled »)	0x0201	0x0202	Sorties accessibles par le maître DeviceNet 510 octets 1 zone de données de sortie	:	0x03FF				
Adresses	Zone des données de Sortie													
0x0200	Mot de commande du maître DeviceNet (sauf si « Control/Status Byte » = « Disabled »)													
0x0201														
0x0202	Sorties accessibles par le maître DeviceNet 510 octets 1 zone de données de sortie													
:														
0x03FF														
<p>Structure de la mémoire de la passerelle LUF9 :</p> <p><b>Données générales</b></p>	<ul style="list-style-type: none"> <li>• 960 octets inaccessibles par le maître DeviceNet.</li> </ul> <table border="1" data-bbox="529 716 1365 1014"> <thead> <tr> <th>Adresses</th> <th>Zone des données Générales</th> </tr> </thead> <tbody> <tr> <td>0x0400</td> <td rowspan="2">Zone d'entrée réservée aux Mailboxes (288 octets)</td> </tr> <tr> <td>0x051F</td> </tr> <tr> <td>0x0520</td> <td rowspan="2">Zone de sortie réservée aux Mailboxes (288 octets)</td> </tr> <tr> <td>0x063F</td> </tr> <tr> <td>0x0640</td> <td rowspan="4">Zone interne réservée à la gestion du réseau amont (384 octets) (zone d'entrée / zone de sortie / zone bidirectionnelle)</td> </tr> <tr> <td>.....</td> </tr> <tr> <td>0x07BF</td> </tr> <tr> <td></td> </tr> </tbody> </table> <p>REMARQUE : Vous pouvez utiliser cette zone de données pour y placer les données d'une réponse Modbus que vous ne souhaitez pas faire remonter jusqu'au maître DeviceNet. Vous pouvez également utiliser cette zone mémoire pour les transferts de données entre des commandes et/ou des transactions étant donné que cette zone est à la fois une zone d'entrée et de sortie. Dans ce cas, utilisez toujours 0x0400 comme adresse de départ. Si vous réutilisez plusieurs fois les adresses dans cette zone, les emplacements mémoire correspondants apparaîtront en rouge dans la section « General Area » de la fenêtre « Sub-network Monitor » (voir exemple page 120). Cependant, cela n'aura aucune conséquence sur le fonctionnement de la passerelle.</p>	Adresses	Zone des données Générales	0x0400	Zone d'entrée réservée aux Mailboxes (288 octets)	0x051F	0x0520	Zone de sortie réservée aux Mailboxes (288 octets)	0x063F	0x0640	Zone interne réservée à la gestion du réseau amont (384 octets) (zone d'entrée / zone de sortie / zone bidirectionnelle)	.....	0x07BF	
Adresses	Zone des données Générales													
0x0400	Zone d'entrée réservée aux Mailboxes (288 octets)													
0x051F														
0x0520	Zone de sortie réservée aux Mailboxes (288 octets)													
0x063F														
0x0640	Zone interne réservée à la gestion du réseau amont (384 octets) (zone d'entrée / zone de sortie / zone bidirectionnelle)													
.....														
0x07BF														
<p>Ordre de transfert des données (<i>swapping</i>)</p>	<ul style="list-style-type: none"> <li>• Réseau DeviceNet : LSB en premier et MSB en dernier.</li> <li>• Réseau Modbus RTU : MSB en premier et LSB en dernier.</li> <li>• Passerelle LUF9 : MSB stocké dans l'adresse mémoire la plus basse.</li> </ul> <p>→ Dans la plupart des cas, l'option qui doit être retenue pour les données Modbus stockées dans la mémoire de la passerelle est « Swap 2 bytes ». Cette option concerne tous les champs « Data », « Preset Data » et « Variable Data » des trames des requêtes et des réponses Modbus.</p>													

## Annexe B : Configuration par défaut

La configuration décrite ci-dessous correspond à la configuration par défaut de la passerelle LUF9.

**REMARQUE :** Ce chapitre est principalement destiné à renseigner l'utilisateur sur les performances obtenues sur le réseau aval Modbus. Il permet à l'utilisateur de décider s'il doit, par exemple, modifier la période des échanges cycliques effectués avec un ou plusieurs des départs-moteurs TeSys U (voir chapitre 6).

### Configuration des échanges Modbus

La passerelle LUF9 effectue quatre types d'échanges avec chacun des 8 départs-moteurs TeSys U. Les deux premiers échanges sont cycliques et permettent de commander et de surveiller le départ-moteur. Les deux derniers échanges sont apériodiques (uniquement sur changement des valeurs des données à transmettre au départ-moteur) et permettent de lire et de modifier la valeur de n'importe quel paramètre du départ-moteur.

Fonction	Fonction Modbus	Nombre d'octets (1)	Echange entre la passerelle LUF9 et le départ-moteur TeSys U
0x03	Read Holding Registers	11,5 + 10,5	Lecture périodique (période de 300 ms) du seul registre d'état du départ-moteur TeSys U (adresse 455 = 0x01C7)
0x10	Preset Multiple Registers	14,5 + 11,5	Ecriture périodique (période de 300 ms) du seul registre d'état du départ-moteur TeSys U (adresse 704 = 0x02C0)
(0x03)	(Read Holding Register)	011,5 + 10,5	Lecture apériodique de la valeur d'un seul paramètre, pour un seul départ-moteur TeSys U à la fois (fonction et adresse fournies par l'utilisateur)
(0x06)	(Preset Single Register)	11,5 + 11,5	Ecriture apériodique de la valeur d'un seul paramètre, pour un seul départ-moteur TeSys U à la fois (fonction, adresse et valeur fournies par l'utilisateur)

(1) Nombre d'octets de la requête (Query) + nombre d'octets de la réponse (Response), avec + 3,5 caractères de temps de silence pour chacune de ces deux trames. Chaque octet sera transmis sous la forme d'un groupe de 10 bits (8 bits de données, 1 bit de start et 1 bit de stop). Ces valeurs permettent de calculer le trafic approximatif sur le réseau aval Modbus de la manière suivante :

Volume du trafic périodique (période de 300 ms)....  $[ ( 11,5 + 10,5 ) + ( 14,5 + 11,5 ) ] \times ( 8 + 1 + 1 ) = 480 \text{ bits}$

Pour 1 départ-moteur TeSys U.....  $1 \times 480 \times ( 1\ 000 \div 300 ) = 01\ 600 \text{ bits/s}$

Pour 8 départs-moteurs TeSys U .....  $8 \times 480 \times ( 1\ 000 \div 300 ) = 12\ 800 \text{ bits/s}$

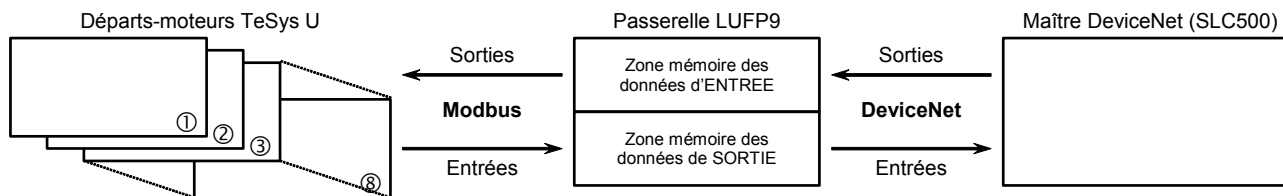
Par conséquent, sur un réseau fonctionnant à 9 600 bits/s, il sera nécessaire d'augmenter de manière importante le temps de cycle de tout ou partie des commandes Modbus périodiques. En revanche, à la vitesse de 19 200 bits/s (vitesse par défaut), la réserve de la bande passante est suffisante pour assurer des communications correctes, même en cas de mode dégradé occasionnel (répétitions de trames par ré-émission), et pour permettre l'utilisation des échanges apériodiques de paramétrage.

# Annexe B : Configuration par défaut

## Contenu de la mémoire DPRAM de la passerelle

La mémoire DPRAM de la passerelle LUF9 contient toutes les données échangées entre la passerelle et les 8 départs-moteurs TeSys U, ainsi que deux registres spéciaux uniquement échangés entre la passerelle et le maître DeviceNet (mots utiles à la gestion du réseau aval Modbus).

Le flux des données échangées entre les départs-moteurs TeSys U, la passerelle et le maître DeviceNet est schématisé ci-dessous, afin de représenter l'implication de la mémoire de la passerelle dans ces échanges :



## Zone mémoire des données d'entrée

La passerelle dispose de 512 octets d'entrée. Seuls les 32 premiers octets sont utilisés. L'ensemble de ces 32 octets forme la zone d'entrée de la passerelle, référencée « Input 1 » dans le configurateur RSNetWorx.

Service	Adresse	Taille	Description
Gestion du réseau aval Modbus	0x0000	1 mot	Mot d'état de la passerelle
Communications périodiques — Surveillance des départs-moteurs TeSys U	0x0002	1 mot	Valeur du registre d'état du départ-moteur ①
	0x0004	1 mot	Valeur du registre d'état du départ-moteur ②
	0x0006	1 mot	Valeur du registre d'état du départ-moteur ③
	0x0008	1 mot	Valeur du registre d'état du départ-moteur ④
	0x000A	1 mot	Valeur du registre d'état du départ-moteur ⑤
	0x000C	1 mot	Valeur du registre d'état du départ-moteur ⑥
	0x000E	1 mot	Valeur du registre d'état du départ-moteur ⑦
	0x0010	1 mot	Valeur du registre d'état du départ-moteur ⑧
—	0x0012	1 octet	Emplacement mémoire libre
Communications aperiodiques — Lecture de la valeur d'un paramètre de départ-moteur (REPONSE)	0x0013	1 octet	N° esclave (0x01 à 0x08)
	0x0014	1 octet	Numéro de la fonction (0x03)
	0x0015	1 octet	Nombre d'octets lus (0x02)
	0x0016	1 mot	Valeur du paramètre lu (0xxxxx)
Communications aperiodiques — Ecriture de la valeur d'un paramètre de départ-moteur (REPONSE)	0x0018	1 octet	N° esclave (0x01 à 0x08)
	0x0019	1 octet	Numéro de la fonction (0x06)
	0x001A	1 mot	Adresse du paramètre écrit (0xxxxx)
	0x001C	1 mot	Valeur du paramètre écrit (0xxxxx)
Communications aperiodiques (« Trigger bytes » des réponses)	0x001E	1 octet	Compteur de réponse de la lecture d'un paramètre
	0x001F	1 octet	Compteur de réponse de l'écriture d'un paramètre
—	0x0020	1 octet	Zone d'entrée libre (480 octets)
	...	...	
	0x01FF	1 octet	

## Annexe B : Configuration par défaut

### Zone mémoire des données de sortie

La passerelle dispose de 512 octets de sortie. Seuls les 32 premiers octets sont utilisés. L'ensemble de ces 32 octets forment la zone de sortie de la passerelle, référencée « Output 1 » dans le configurateur RSNetWorx.

Service	Adresse	Taille	Description
Gestion du réseau aval Modbus	0x0200	1 mot	Mot de commande du maître DeviceNet
Communications périodiques — Commande des départs-moteurs TeSys U	0x0202	1 mot	Valeur du registre de commande du départ-moteur ①
	0x0204	1 mot	Valeur du registre de commande du départ-moteur ②
	0x0206	1 mot	Valeur du registre de commande du départ-moteur ③
	0x0208	1 mot	Valeur du registre de commande du départ-moteur ④
	0x020A	1 mot	Valeur du registre de commande du départ-moteur ⑤
	0x020C	1 mot	Valeur du registre de commande du départ-moteur ⑥
	0x020E	1 mot	Valeur du registre de commande du départ-moteur ⑦
	0x0210	1 mot	Valeur du registre de commande du départ-moteur ⑧
Communications aperiodiques — Lecture de la valeur d'un paramètre de départ-moteur (REQUETE)	0x0212	1 octet	N° esclave (0x01 à 0x08)
	0x0213	1 octet	Numéro de la fonction (0x03)
	0x0214	1 mot	Adresse du paramètre à lire (0xxxxx)
	0x0216	1 mot	Nombre de paramètres à lire (0x0001)
Communications aperiodiques — Ecriture de la valeur d'un paramètre de départ-moteur (REQUETE)	0x0218	1 octet	N° esclave (0x01 à 0x08)
	0x0219	1 octet	Numéro de la fonction (0x06)
	0x021A	1 mot	Adresse du paramètre à écrire (0xxxxx)
	0x021C	1 mot	Valeur du paramètre à écrire (0xxxxx)
Communications aperiodiques (« Trigger bytes » des requêtes)	0x021E	1 octet	Compteur de requête de la lecture d'un paramètre
	0x021F	1 octet	Compteur de requête de l'écriture d'un paramètre
—	0x0220 ... 0x03FF	1 octet ... 1 octet	Zone de sortie libre (480 octets)

# Annexe B : Configuration par défaut

---

## Nombre total de requêtes et de réponses Modbus

**Le nombre total de requêtes et de réponses Modbus est égal à 36** (2 requêtes et 2 réponses périodiques pour chacun des 8 départs-moteurs TeSys U, plus 2 requêtes et 2 réponses aperiodiques pour l'ensemble de ces départs-moteurs). Puisque le nombre total de requêtes et de réponses Modbus qu'il est possible de configurer pour une seule et même passerelle est limité à 100, il reste une réserve de 64 requêtes et réponses Modbus (c'est-à-dire l'équivalent de 32 commandes Modbus).

Cette réserve permet donc d'ajouter jusqu'à 4 commandes Modbus pour chacun des 8 départs-moteurs TeSys U, puisque cet ajout nécessiterait l'utilisation de 64 requêtes et réponses Modbus (4 fois 1 requête et 1 réponse pour chacun des 8 départs-moteurs, c.-à-d.  $4 \times (1+1) \times 8$ ).

# Annexe C : Exemple d'utilisation (RSLogix 500)

**REMARQUE** : Cette annexe est réservée aux utilisateurs possédant une bonne connaissance des produits Rockwell Automation RSNetWorx et RSLogix 500.

Un exemple d'utilisation est disponible. Il est composé de deux fichiers. Le premier de ces fichiers, nommé « SLC\_Guide\_LUFP9.dnt », présente la configuration du scanner DeviceNet sous RSNetWorx, décrit dans les chapitres précédents. Le second, nommé « SLC\_Guide\_LUFP9\_EN.rss », est un fichier RSLogix 500 et constitue donc l'exemple en lui-même.

La configuration du fichier RSNetWorx correspondant exactement à ce qui est décrit dans les chapitres précédents, son contenu ne sera pas repris ici. En revanche, le fichier RSLogix 500 est décrit ci-après, en se basant sur la structure des sous-programmes utilisés.

## Programme principal : « LAD 2 - MAIN\_LUFP9 »

Le rôle du programme principal consiste à activer les communications DeviceNet et Modbus, ainsi qu'à appeler les autres sous-programmes, décrits dans les chapitres suivants. Les processus effectués dans le programme principal sont décrits ci-dessous, dans l'ordre de leur exécution :

- Validation des échanges DeviceNet du scanner par activation du bit O:1.0/0.
- Activation des communications Modbus de la passerelle à l'aide des bits 13 (FB\_DU) et 14 (FB\_HS\_SEND) du mot de commande du maître DeviceNet. Ces deux bits correspondent aux bits O:1.1/5 et O:1.1/6 du scanner DeviceNet.

**REMARQUE** : Ce processus n'est utile qu'à la condition que l'option « Control/Status Byte » soit définie sur « Enabled ». Dans le cas de la configuration par défaut de la passerelle LUFP9 (« Control/Status Byte » = « Enabled but no startup lock »), ce processus est inutile mais peut tout de même être conservé. *Enfin, cet exemple ne doit pas être utilisé lorsque cette option est définie sur « Disabled », car les mots I:1.1 et O:1.1 ne sont plus réservés à la « gestion du réseau aval Modbus ».* Reportez-vous au chapitre 5, pour de plus amples informations à ce sujet.

- Acquiescement automatique des diagnostics de la passerelle par le maître DeviceNet. Il suffit de recopier la valeur du bit 15 (ABC\_HS\_SEND) du mot d'état de la passerelle dans le bit 15 (FB\_HS\_CONFIRM) du mot de commande du maître DeviceNet (voir chapitre 5). Cet acquiescement automatique permet surtout de ne pas bloquer le mécanisme de remontée des diagnostics de la passerelle au maître DeviceNet.
- Commande/surveillance du départ-moteur « TeSys U n°1 » par utilisation du sous-programme U:3, c'est-à-dire le sous-programme « LAD 3 - CMD\_SURV ». Ce sous-programme utilise des variables locales en tant que paramètres. Le mot N7:0 est utilisé pour indexer le registre de sortie et le registre d'entrée, utilisés pour effectuer la commande et la surveillance du départ-moteur « TeSys U n°1 ». Avant l'appel du sous-programme, la valeur de ce mot est donc fixée à 2 afin d'accéder aux mots O:1.2 et I:1.2. N7:0 est également utilisé pour indexer l'un des bits de chacun des registres N7:32, 33, 34 et 35 (registres manipulés par l'utilisateur).
- Commande/surveillance du départ-moteur « TeSys U n°2 » : *Idem*, mais en fixant la valeur de N7:0 à 3 (O:1.3 et I:1.3).
- Commande/surveillance du départ-moteur « TeSys U n°3 » : *Idem*, mais en fixant la valeur de N7:0 à 4 (O:1.4 et I:1.4).
- Commande/surveillance du départ-moteur « TeSys U n°4 » : *Idem*, mais en fixant la valeur de N7:0 à 5 (O:1.5 et I:1.5).
- Commande/surveillance du départ-moteur « TeSys U n°5 » : *Idem*, mais en fixant la valeur de N7:0 à 6 (O:1.6 et I:1.6).
- Commande/surveillance du départ-moteur « TeSys U n°6 » : *Idem*, mais en fixant la valeur de N7:0 à 7 (O:1.7 et I:1.7).

## Annexe C : Exemple d'utilisation (RSLogix 500)

- Commande/surveillance du départ-moteur « TeSys U n°7 » : *Idem*, mais en fixant la valeur de N7:0 à 8 (O:1.8 et I:1.8).
- Commande/surveillance du départ-moteur« TeSys U n°8 » : *Idem*, mais en fixant la valeur de N7:0 à 9 (O:1.9 et I:1.9).
- Lecture de la valeur d'un même paramètre sur l'ensemble des départs-moteurs TeSys U, par utilisation du sous-programme U:4, c'est-à-dire le sous-programme « LAD 4 - LECT\_PAR ».
- Ecriture de la valeur d'un paramètre dans un seul départ-moteur TeSys U à la fois, par utilisation du sous-programme U:5, c'est-à-dire le sous-programme « LAD 5 - LECT\_PAR ».
- Mise à jour de la sortie O:1.16 à l'aide des deux compteurs N7:36 et N7:37. Cette sortie contient les deux « Trigger bytes » utilisés pour provoquer l'émission de la requête de lecture d'un paramètre (LSB) et la requête d'écriture d'un paramètre (MSB). Ces deux compteurs sont mis à jour de manière indépendante dans les sous-programmes « LAD 4 – RD\_PAR », pour N7:36, et « LAD 5 – WR\_PAR », pour N7:37.

**REMARQUE** : La lecture d'un paramètre sur tous les départs-moteurs et l'écriture d'un paramètre sur l'un d'entre eux peuvent être effectuées en parallèle, car ces services utilisent des commandes Modbus différentes.

Les différentes données utilisées par le programme principal sont rassemblées dans le tableau suivant :

Adresse	Symbole	Description
I:1.1/07 → I:1/23	ABC_HS_SEND	Bascule indiquant la présence d'un nouveau diagnostic de la passerelle
O:1.0/00 → O:1/00	VALIDATION_DU_SCAN	Validation des communications DeviceNet: ce bit doit être à 1 pour valider les échanges
O:1.1/05 → O:1/21	FB_DU	Activation des communications Modbus par la passerelle
O:1.1/06 → O:1/22	FB_HS_SEND	Bascule indiquant à la passerelle la présence d'une nouvelle commande
O:1.1/07 → O:1/23	FB_HS_CONFIRM	Bit d'acquiescement des diagnostics de la passerelle par le maître DeviceNet
N7:0	MODULE	Paramètre d'accès (index) au départ-moteur (appelé « module » pour simplifier)
O:1.16	TRIGGER_OUT_RD_WR	« Trigger bytes » provoquant l'émission de la requête de lecture (LSB) ou d'écriture (MSB) d'un paramètre
N7:36	_____	Compteur local pour le « trigger byte » de la requête de lecture d'un paramètre
N7:37	_____	Compteur local pour le « trigger byte » de la requête d'écriture d'un paramètre



# Annexe C : Exemple d'utilisation (RSLogix 500)

---

## Sous-programme de commande/surveillance d'un départ-moteur TeSys U : « LAD 3 - CMD\_SURV »

Le rôle de ce sous-programme consiste à effectuer une commande très simple sur l'un des départs-moteurs TeSys U, en fonction de son état actuel et des commandes de l'utilisateur. Les processus effectués dans ce sous-programme sont décrits ci-dessous, dans l'ordre de leur exécution :

- Commande du moteur en marche avant / marche arrière / arrêt. Le registre N7:0 est utilisé en guise de paramètre. Il contient le numéro du mot d'entrée et du mot de sortie à utiliser pour commander et pour surveiller le départ-moteur TeSys U. Ce même numéro sert à indexer le bit à prendre en compte dans les registres N7:32 à N7:35. Le mot d'entrée utilisé est compris entre I:1.2 et I:1.9 (départs-moteurs n°1 à 8), et le mot de sortie utilisé est compris entre O:1.2 et O:1.9 (*idem*). La valeur de N7:0 doit donc être comprise entre 2 et 9, selon le numéro du départ-moteur commandé.

L'utilisateur pilote le mode de marche du départ-moteur à l'aide des bits 2 à 9 (départs-moteurs n°1 à 8) des registres N7:32 ( Run (1) / Stop (0) ) et N7:33 ( Marche Avant (0) / Marche Arrière (1) ).

Les commandes de marche avant, de marche arrière et d'arrêt du départ-moteur TeSys U sont effectuées aux conditions suivantes :

- Bit 14 du mot d'état d'un TeSys U = 0      Le départ-moteur n'est pas en mode local.
- Bit 2 du mot d'état d'un TeSys U = 0      Le départ-moteur n'est pas en défaut.
- Bit 0 ou 1 du mot d'état d'un TeSys U = 1    Le départ-moteur est en état « Ready » ou « Switched on ».

Lorsque ces conditions sont toutes rassemblées, les registres N7:32 et N7:33 (bit 2 à 9, selon la valeur de N7:0) sont utilisés pour commander soit la marche avant / marche arrière du départ-moteur, soit son arrêt par freinage. Ces deux registres sont mis à jour bit à bit par l'utilisateur, en fonction des commandes qu'il désire effectuer.

- Remise à zéro des défauts du départ-moteur TeSys U. Le registre N7:0 est utilisé de la même manière que ci-dessus et les mots d'entrée et de sortie sont les mêmes que pour la commande du départ-moteur.

Lorsqu'un défaut est présent sur le départ-moteur (bit 2 du registre de surveillance égal à 1), ce défaut est recopié dans l'un des bits 2 à 9 (un bit par départ-moteur) du registre N7:34 ( Présence Défaut (1) / Départ-moteur OK (0) ), simplement pour faire figurer cet état de manière conjointe à la commande utilisateur qui permet de remettre à zéro les défauts du départ-moteur. Cette commande utilisateur correspond à l'un des bits 2 à 9 du registre N7:35 ( RAZ défaut (1) ) et sert à activer le bit 3 du registre de commande du départ-moteur TeSys U correspondant (bit de « Reset »), c'est-à-dire le bit O:1.[N7:0]/3.

Cette commande de remise à zéro des défauts par l'utilisateur est ensuite annulée par le programme lorsque le départ-moteur TeSys U ne signale plus la présence d'un défaut.

## Annexe C : Exemple d'utilisation (RSLogix 500)

Les différentes données utilisées par ce sous-programme sont rassemblées dans le tableau suivant :

Adresse	Symbole	Description
I:1.[N7:0]/00	—	Bit 00 « Ready » du registre d'état du TeSys U
I:1.[N7:0]/01	—	Bit 01 « On » du registre d'état du TeSys U
I:1.[N7:0]/02	—	Bit 02 « Fault » du registre d'état du TeSys U
I:1.[N7:0]/14	—	Bit 14 « Reserved : Local control » du registre d'état du départ-moteur TeSys U
N7:32/[N7:0]	CMD_RUN [ MODULE ]	<b>Commande utilisateur</b> : Marche (1) / Arrêt (0) sur le départ-moteur dont le numéro est défini sur N7:0
N7:33/[N7:0]	CMD_REVERSE [ MODULE ]	<b>Commande utilisateur</b> : Avant (0) / Arrière (1) sur le départ-moteur dont le numéro est défini sur N7:0
N7:34/[N7:0]	SURV_FAULTY_DEV [ MODULE ]	<b>Surveillance utilisateur</b> : Présence (1) / Absence (0) d'un défaut sur le départ-moteur dont le numéro est défini sur N7:0
N7:35/[N7:0]	CMD_RESET [ MODULE ]	<b>Commande utilisateur</b> : RAZ défaut (1) sur le départ-moteur dont le numéro est défini sur N7:0
O:1.[N7:0]/00	—	Bit 0 « Reserved : Run Forward » du registre de commande du TeSys U repéré à l'aide de l'index N7:0
O:1.[N7:0]/01	—	Bit 1 « Reserved : Run Reverse » du registre de commande du TeSys U repéré à l'aide de l'index N7:0
O:1.[N7:0]/02	—	Bit 2 « Reserved (stopping) » du registre de commande du TeSys U repéré à l'aide de l'index N7:0
O:1.[N7:0]/03	—	Bit 3 « Reset » du registre de commande du TeSys U repéré à l'aide de l'index N7:0
N7:0	MODULE	Paramètre d'accès au départ-moteur (index compris entre 2 et 9, pour les départs-moteurs TeSys U n°1 à n°8)

L'exemple comporte un écran de surveillance des données personnalisé, appelé « CDM 0 - CMD\_SURV », afin de simplifier l'utilisation de cet exemple. Le contenu de cet écran est reproduit ci-dessous :

Adresse	Symbole	Affichage
O:1/00	VALIDATION_DU_SCAN	Binaire
O:1/21	FB_DU	Binaire
O:1/22	FB_HS_SEND	Binaire
N7:0	MODULE	Décimal
N7:32	CMD_RUN	Binaire
N7:33	CMD_REVERSE	Binaire
N7:34	SURV_FAULTY_DEV	Binaire
N7:35	CMD_RESET	Binaire
I:1.2	SURV_TESYS_U_1	Binaire
O:1.2	CMD_TESYS_U_1	Binaire
I:1.3	SURV_TESYS_U_2	Binaire
O:1.3	CMD_TESYS_U_2	Binaire

Adresse	Symbole	Affichage
I:1.4	SURV_TESYS_U_3	Binaire
O:1.4	CMD_TESYS_U_3	Binaire
I:1.5	SURV_TESYS_U_4	Binaire
O:1.5	CMD_TESYS_U_4	Binaire
I:1.6	SURV_TESYS_U_5	Binaire
O:1.6	CMD_TESYS_U_5	Binaire
I:1.7	SURV_TESYS_U_6	Binaire
O:1.7	CMD_TESYS_U_6	Binaire
I:1.8	SURV_TESYS_U_7	Binaire
O:1.8	CMD_TESYS_U_7	Binaire
I:1.9	SURV_TESYS_U_8	Binaire
O:1.9	CMD_TESYS_U_8	Binaire

# Annexe C : Exemple d'utilisation (RSLogix 500)

## Sous-programme de lecture d'un paramètre dans tous les départs-moteurs TeSys U : « LAD 4 - LECT\_PAR »

Le rôle de ce sous-programme consiste à effectuer la lecture de la valeur d'un seul et même paramètre pour l'ensemble des départs-moteurs TeSys U. Les résultats sont placés, au fur et à mesure de la lecture, dans un tableau commençant en N7:4 (départ-moteur n°1) et s'achevant en N7:11 (départ-moteur n°8). L'index N7:2 est utilisé pour avoir accès à ces différentes adresses. Les processus effectués dans ce sous-programme sont décrits ci-dessous, dans l'ordre de leur exécution :

- La modification par l'utilisateur du numéro (ou adresse) du paramètre à lire (N7:1) provoque l'initialisation des données utilisées par le sous-programme, mais uniquement si le processus de lecture précédent est achevé (B3:0/0 = 0). La comparaison entre N7:1 (nouvelle adresse) et O:1.11 (adresse dans la dernière commande utilisée) est effectuée au travers d'une variable de travail, N9:0, dans laquelle l'inversion entre le LSB et le MSB de la nouvelle adresse est effectuée. Les initialisations sont résumées ci-dessous :
  - B3:0/0 = 1 ..... Lecture d'un paramètre sur tous les départs-moteurs TeSys U : En cours de réalisation.
  - Reset (C5:0) ..... Réinitialisation du compteur du nombre de départs-moteurs interrogés.
  - Reset (T4:0)..... Réinitialisation de la temporisation associée au timeout d'attente d'une réponse de lecture d'un paramètre.
  - N7:2 = 4 ..... Indice dans le tableau des résultats → N° du 1<sup>er</sup> élément du tableau = N7:4.
  - N7:3 = 1 ..... Adresse de l'esclave Modbus interrogé → Adresse du premier départ-moteur TeSys U, c'est-à-dire 1.
  - N7:[4..11] = 0 ..... RAZ du contenu du tableau des résultats.
  - B3:0/5 = 0 ..... Autorisation de mise à jour du compteur / « trigger byte » provoquant l'émission de la requête.
- Mise à jour des données de sortie qui correspondent à la requête de lecture (O:1.10 à O:1.12), incrémentation du compteur N7:36 (« trigger byte ») d'une unité. Cette mise à jour n'est effectuée qu'une seule fois (utilisation du bit B3:0/5 dans ce but). **Rappel** : Dans la configuration par défaut de la passerelle LUF9, ces données de sortie correspondent aux données de la commande Modbus personnalisée « Transactions 1 » du nœud « TeSys U n°1 ». La trame de la requête de cette commande personnalisée est envoyée sur changement de la valeur du « trigger byte » situé dans les bits 0-7 du mot O:1.16 (« Update mode » = « Change of state on trigger »). Par conséquent, l'incrémentation du compteur N7:36, puis la mise à jour de O:1.16 à l'aide de N7:36 (dans « LAD 2 – MAIN\_LUF9 »), provoque l'envoi de cette requête. L'ensemble des données de sortie O:1.10 à O:1.12 doivent être correctes pour que le contenu de la requête Modbus soit cohérent !
- Vérification des données de la réponse Modbus qui correspond à cette commande de lecture. Les valeurs des entrées I:1.10 et I:1.11 sont comparées à celles de la sortie O:1.10 et de la valeur 0x02xx (masque ET égal à 0xFF00) afin de déterminer si la réponse à la commande est arrivée ou non. Si le numéro de l'esclave et le numéro de la fonction correspondent à ceux de la requête (voir ci-dessus) et si le nombre d'octets de données reçus est correct, le bit B3:0/1 est activé pour signaler au reste du sous-programme que la réponse est arrivée et qu'elle est correcte. La variable temporaire N9:0 est utilisée pour comparer les entrées et les sorties sous un même format.
- Recopie de la valeur du paramètre lu dans le tableau des résultats. La valeur de I:1.12 est donc transférée à l'emplacement qui est réservé au résultat du départ-moteur actuellement interrogé (utilisation de l'index N7:2). Ce transfert n'a lieu qu'à la condition que la réponse soit arrivée et que son contenu soit correct (bit B3:0/1 actif). Le LSB et le MSB de cette valeur sont ensuite permutés dans ce tableau afin de rétablir la valeur du paramètre lu. La temporisation du timeout d'attente de la réponse (T4:0) est réinitialisée pour préparer le processus de la lecture du même paramètre sur le départ-moteur suivant.
- Gestion du timeout d'attente de la réponse (bloc TON sur la variable T4:0). Tant que la réponse n'est pas arrivée ou que son contenu est incorrect (bit B3:0/1 = 0), une temporisation de 3 secondes est activée. Sur déclenchement de ce timeout (T4:0/DN = 1), la temporisation associée est réinitialisée et un résultat égal à -1 est placé dans le tableau des résultats, à l'emplacement normalement réservé au départ-moteur interrogé.
- Sur réception de la réponse, ou suite au déclenchement du timeout, les données internes au sous-programme sont mises à jour pour permettre la lecture du même paramètre sur le départ-moteur suivant, et ce jusqu'au dernier départ-moteur, pour un total de 8 départs-moteurs (adresses 1 à 8). Le compteur C5:0 permet de compter le nombre de départs-moteurs ayant été interrogés jusqu'à présent.

## Annexe C : Exemple d'utilisation (RSLogix 500)

- Lorsque la lecture du 8<sup>ème</sup> départ-moteur est achevée (compteur C5:0 atteignant sa valeur de présélection), le processus de lecture est stoppé (RAZ du bit B3:0/0). En revanche, tant que la lecture du paramètre du 8<sup>ème</sup> départ-moteur n'est pas achevée, le sous-programme recommence le cycle automate suivant depuis le début (passage au départ-moteur suivant ou poursuite de l'attente de la réponse pour le départ-moteur actuellement interrogé).

Les différentes données utilisées par ce sous-programme sont rassemblées dans le tableau suivant :

Adresse	Symbole	Description
B3:0/0	LECT_RUNNING	Lecture d'un paramètre dans tous les départs-moteurs TeSys U : En cours
B3:0/1	LECT_OK_KO	Lecture d'un paramètre dans tous les départs-moteurs TeSys U : Lecture correcte (OK) ou incorrecte (KO) pour un départ-moteur (si la réponse est arrivée ou sur déclenchement du timeout T4:0)
B3:0/5	————	Mise à jour du « trigger byte » d'émission effectuée : Oui (1) / Non (0)
C5:0	CPT_LECT_TESYS_U	Lecture d'un paramètre dans les départs-moteurs TeSys U : Compteur. Lorsque la valeur de ce compteur parvient à 9, le processus de lecture d'un paramètre sur l'ensemble des départs-moteurs TeSys U est arrêté.
I:1.10	CR_RDPAR_XXX_SLAVE	Résultat de la lecture d'un paramètre : Esclave (0x01 à 0x08) en MSB. La valeur de ce champ est comparée à celle du champ correspondant dans la trame de la requête. Le LSB de ce mot d'entrée n'est pas utilisé.
I:1.11	CR_RDPAR_FCT_BYTES	Résultat de la lecture d'un paramètre : Fonction (toujours 0x03) en LSB (la valeur de ce champ est comparée à celle du champ correspondant dans la trame de la requête) + Nombre d'octets lus (0x02) en MSB (valeur masquée et vérifiée).
I:1.12	CR_RDPAR_VALUE	Résultat de la lecture d'un paramètre : Valeur du paramètre lu (MSB et LSB inversés). Cette valeur est placée dans le tableau N7:[N7:2], puis son MSB et son LSB y sont permutés afin de rétablir la valeur correcte du paramètre lu.
N7:1	NUMPARAM	<b>Commande utilisateur</b> : Numéro du paramètre lu.
N7:2	LECT_INDEX	Indice dans le tableau des résultats de la lecture d'un paramètre TeSys U. Valeur = 4 à 11 (départs-moteurs n°1 à 8).
N7:3	ADRESSE	Adresse de l'esclave Modbus dont l'un des paramètres est en cours de lecture. Valeur = 1 à 8.
N7:[N7:2]	— [LECT_INDEX]	Tableau des résultats de la lecture d'un paramètre TeSys U (départs-moteurs n°1 à 8). Eléments N7:4 à N7:11 (voir N7:2). Valeur = -1 en cas d'erreur (timeout de l'attente de la réponse).
N7:36	————	Compteur local correspondant au « trigger byte » de la requête de lecture.
N9:0	VAR_EPHEMERE_1	Variable temporaire de travail servant à effectuer des calculs intermédiaires.
O:1.10	RDPAR_SLAVE_FCT	Demande de lecture d'un paramètre : Esclave (de 0x01 à 0x08) en LSB + Fonction (toujours 0x03) en MSB.
O:1.11	RDPAR_ADRPAR	Demande de lecture d'un paramètre : Adresse du paramètre (recopie de N7:1, avec permutation du LSB et du MSB).
O:1.12	RDPAR_NBPARS	Demande de lecture d'un paramètre : Nombre de paramètres à lire (toujours 0x0001, mais avec inversion du MSB et du LSB, c'est-à-dire 0x0100).
T4:0	TIMEOUT_LECT_PARAM	Temporisation du timeout de la lecture d'un paramètre (3 secondes)

L'exemple comporte un écran de surveillance personnalisée des données, appelé « CDM 1 - LECT\_PAR », afin de simplifier l'utilisation de cet exemple. Le contenu de cet écran est reproduit ci-dessous :

Adresse	Symbole	Affichage
N7:1	NUMPARAM	Décimal
B3:0/0	LECT_RUNNING	Binaire
B3:0/1	LECT_OK_KO	Binaire
N7:2	LECT_INDEX	Décimal
N7:3	ADRESSE	Décimal
N7:4	PARLU1	Décimal
N7:5	PARLU2	Décimal
N7:6	PARLU3	Décimal
N7:7	PARLU4	Décimal
N7:8	PARLU5	Décimal
N7:9	PARLU6	Décimal

Adresse	Symbole	Affichage
N7:10	PARLU7	Décimal
N7:11	PARLU8	Décimal
O:1.10	RDPAR_SLAVE_FCT	Hexadécimal
O:1.11	RDPAR_ADRPAR	Décimal
O:1.12	RDPAR_NBPARS	Hexadécimal
I:1.10	CR_RDPAR_XXX_SLAVE	Hexadécimal
I:1.11	CR_RDPAR_FCT_BYTES	Hexadécimal
I:1.12	CR_RDPAR_VALUE	Hexadécimal
I:1.16	TRIGGER_IN_RD_WR	Hexadécimal
O:1.16	TRIGGER_OUT_RD_WR	Hexadécimal
N7:36	————	Hexadécimal
B3:0/5	————	Binaire

## Annexe C : Exemple d'utilisation (RSLogix 500)

### Sous-programme d'écriture d'un paramètre dans un seul départ-moteur TeSys U : « LAD 5 - ECRIT\_PAR »

Le rôle de ce sous-programme consiste à effectuer l'écriture de la valeur d'un paramètre dans un seul départ-moteur TeSys U. L'utilisateur doit procéder à la saisie de l'adresse du départ-moteur TeSys U (N7:12), de l'adresse du paramètre (N7:13) et de la valeur à affecter au paramètre (N7:14). Enfin, il doit activer le bit B3:0/2 pour activer le processus d'écriture. Ce bit est automatiquement remis à zéro par le sous-programme LAD 5. Lorsque le processus d'écriture s'achève, le résultat de l'écriture (adresse du paramètre et valeur du paramètre) est placé dans un tableau commençant en N7:16 (pour le départ-moteur n°1) et s'achevant en N7:31 (pour le départ-moteur n°8), en utilisant la variable N7:15 comme index. Deux cases successives de ce tableau sont utilisées pour chaque départ-moteur : La première reçoit l'adresse du paramètre et la deuxième reçoit sa valeur. Les processus effectués dans ce sous-programme sont décrits ci-dessous, dans l'ordre de leur exécution :

- Mise en attente du sous-programme. Tant que l'utilisateur n'a pas activé le bit B3:0/2, le reste du sous-programme n'est pas exécuté. Cela permet à l'utilisateur de saisir les valeurs des données N7:12, 13 et 14 les unes après les autres.
- Initialisation des données que le sous-programme utilise par la suite, mais uniquement si le processus d'écriture précédent est achevé (B3:0/3 = 0). Ces initialisations sont résumées ci-dessous :
  - B3:0/2 = 0 ..... **Commande utilisateur** : RAZ de la commande d'écriture d'un paramètre dans un départ-moteur TeSys U.
  - B3:0/3 = 1 ..... Ecriture d'un paramètre dans un départ-moteur TeSys U : En cours de réalisation.
  - Reset (T4:1) ..... RAZ de la temporisation associée au timeout d'attente d'une réponse d'écriture d'un paramètre.
  - N7:15 = (N7:12 × 2) + 14 ..... Indice dans le tableau des résultats.
  - N7:[N7:15] = { 0 ; 0 } ..... RAZ du contenu du tableau des résultats, mais uniquement pour le départ-moteur concerné par la requête d'écriture (deux octets successifs).
  - B3:0/6 = 0 ..... Autorisation de mise à jour du compteur / « trigger byte » provoquant l'émission de la requête.
- Mise à jour des données de sortie qui correspondent à la requête d'écriture (O:1.13 à O:1.15), incrémentation du compteur N7:37 (« trigger byte ») d'une unité. Cette mise à jour n'est effectuée qu'une seule fois (utilisation du bit B3:0/6 dans ce but).  
**REMARQUE** : Dans la configuration par défaut de la passerelle LUF9, ces données de sortie correspondent aux données de la commande Modbus personnalisée « Transactions 2 » du nœud « TeSys U n°1 ». La trame de la requête de cette commande personnalisée est envoyée sur changement de la valeur du « trigger byte » situé dans les bits 8-15 du mot O:1.16 (« Update mode » = « Change of state on trigger »). Par conséquent, l'incrémentation du compteur N7:37, puis la mise à jour de O:1.16 à l'aide de N7:37 (dans « LAD 2 – MAIN\_LUF9 »), provoque l'envoi de cette requête. L'ensemble des données de sortie O:1.13 à O:1.15 doivent être correctes pour que le contenu de la requête Modbus soit cohérent ! Le LSB et le MSB des sorties O:1.14 et O:1.15 doivent être permutés. La variable temporaire N9:0 est utilisée pour effectuer cette inversion entre les variables N7:13 et N7:14 et les sorties O:1.14 et O:1.15.
- Vérification des données de la réponse Modbus qui correspond à cette commande d'écriture. Les valeurs des entrées I:1.13 à I:1.15 sont comparées à celles des sorties O:1.13 à O:1.15 afin de déterminer si la réponse à la commande est arrivée ou non. Si le numéro de l'esclave, le numéro de la fonction, l'adresse du paramètre et sa valeur correspondent à ceux de la requête (voir ci-dessus) et si le nombre d'octets de données reçus est correct, le bit B3:0/4 est activé pour signaler au reste du sous-programme que la réponse est arrivée et qu'elle est correcte.
- Recopie de l'adresse et de la valeur du paramètre dans le tableau des résultats. Cette recopie est effectuée dans deux emplacements successifs du tableau (indexation effectuée à l'aide de N7:15), réservés au départ-moteur actuellement interrogé, et n'a lieu qu'à la condition que la réponse soit arrivée et que son contenu soit correct (bit B3:0/4 actif). Le LSB et le MSB de chacune de ces deux données sont ensuite permutés afin de rétablir sa valeur correcte. La temporisation du timeout d'attente de la réponse (T4:1) est réinitialisée pour préparer une future commande d'écriture. Le bit B3:0/3 est remis à zéro pour signaler que la commande est achevée, évitant ainsi au reste du sous-programme d'être exécuté.

## Annexe C : Exemple d'utilisation (RSLogix 500)

- Gestion du timeout d'attente de la réponse (T4:1). Tant que la réponse n'est pas arrivée ou que son contenu est incorrect (bit B3:0/4 = 0), une temporisation de 3 secondes est activée. Sur déclenchement de ce timeout (T4:1/DN = 1), la temporisation est réinitialisée, l'adresse du paramètre (O:1.14, après inversion LSB / MSB via la variable de travail N9:0) et une valeur erronée (N9:1 = -1) sont placées dans le tableau des résultats. Cette copie est effectuée dans deux emplacements successifs du tableau, réservés au départ-moteur actuellement interrogé. Enfin, le processus d'écriture est stoppé (RAZ du bit B3:0/3).

Les différentes données utilisées par ce sous-programme sont rassemblées dans le tableau suivant :

Adresse	Symbole	Description
B3:0/2	ECRIT_COMMANDE	<b>Commande utilisateur</b> : Ecriture d'un paramètre dans un départ-moteur TeSys U : l'activation de ce bit est effectuée par l'utilisateur et sa remise à zéro est effectuée par le programme.
B3:0/3	ECRIT_RUNNING	Ecriture d'un paramètre dans un départ-moteur TeSys U : En cours
B3:0/4	ECRIT_OK	Ecriture d'un paramètre dans un départ-moteur TeSys U : Ecriture OK (si la réponse est arrivée et qu'elle est correcte)
B3:0/6	_____	Mise à jour du « trigger byte » d'émission effectuée : Oui (1) / Non (0)
I:1.13	CR_WRPAR_SLAVE_FCT	Résultat de l'écriture de la valeur d'un paramètre : Esclave (de 0x01 à 0x08) en LSB + Fonction (toujours 0x06) en MSB. Les valeurs de ces champs sont comparées à celles de la requête.
I:1.14	CR_WRPAR_ADRPAR	Résultat de l'écriture de la valeur d'un paramètre : Adresse du paramètre. La valeur de ce champ est comparée à celle de la requête (inversion du MSB et du LSB dans le cas de chacun de ces deux champs).
I:1.15	CR_WRPAR_VALUE	Résultat de l'écriture de la valeur d'un paramètre : Valeur du paramètre écrit. La valeur de ce champ est comparée à celle de la requête (inversion du MSB et du LSB dans le cas de chacun de ces deux champs).
N7:12	ECRIT_ESCLAVE	<b>Commande utilisateur</b> : Adresse Modbus du départ-moteur auquel la demande d'écriture doit être envoyée.
N7:13	ECRIT_ADRESSE	<b>Commande utilisateur</b> : Adresse du paramètre <b>REMARQUE</b> : N'essayez pas de modifier la valeur du registre 704 (registre de commande), car celui-ci est déjà commandé par le maître DeviceNet (voir sous-programme « LAD 3 - CMD_SURV ») !
N7:14	ECRIT_VALEUR	<b>Commande utilisateur</b> : Nouvelle valeur du paramètre
N7:15	ECRIT_INDEX	Index dans le tableau des résultats des écritures de paramètres TeSys U (départs-moteurs n°1 à 8). Valeur = $16 + 2 \times (\text{n}^\circ \text{ départ-moteur} - 1) = 16 \text{ à } 30$
N7:[N7:15]	— [ ECRIT_INDEX ]	Tableau des résultats des écritures de paramètres TeSys U (départs-moteurs n°1 à 8). Eléments N7:16 à N7:31 organisés par couple « adresse du paramètre » / « valeur du paramètre », chaque couple occupant deux adresses successives. « Valeur du paramètre » = -1 en cas d'erreur (timeout de l'attente de la réponse).
N7:37	_____	Compteur local correspondant au « trigger byte » de la requête de lecture.
N9:0 N9:1	VAR_EPHEMERE_1 VAR_EPHEMERE_2	Variables temporaires de travail servant à effectuer des calculs intermédiaires (principalement des inversions LSB / MSB).
O:1.13	WRPAR_SLAVE_FCT	Demande d'écriture de la valeur d'un paramètre : Esclave (recopie de N7:12) en LSB + Fonction (toujours 0x06) en MSB.
O:1.14	WRPAR_ADRPAR	Demande d'écriture de la valeur d'un paramètre : Adresse du paramètre (recopie de N7:13, avec permutation du LSB et du MSB).
O:1.15	WRPAR_VALUE	Demande d'écriture de la valeur d'un paramètre : Valeur du paramètre à écrire (recopie de N7:14, avec permutation du LSB et du MSB).

## Annexe C : Exemple d'utilisation (RSLogix 500)

Adresse	Symbole	Description
S:24	INDEX_SYS	Registre d'index utilisé dans les adressages indexés (préfixe : '#')
T4:1	TIMEOUT_ECRIT_PARAM	Temporisation du timeout de l'écriture d'un paramètre (3 secondes)

L'exemple comporte un écran de surveillance personnalisée des données, appelé « CDM 2 - ECRIT\_PAR », afin d'en simplifier l'utilisation. Le contenu de cet écran est reproduit ci-dessous :

Adresse	Symbole	Affichage
N7:12	ECRIT_ESCLAVE	Décimal
N7:13	ECRIT_ADRESSE	Décimal
N7:14	ECRIT_VALEUR	Décimal
B3:0/2	ECRIT_COMMANDE	Binaire
B3:0/3	ECRIT_RUNNING	Binaire
B3:0/4	ECRIT_OK	Binaire
N7:15	ECRIT_INDEX	Décimal
N7:16	PARECRIT_1_ADRESSE	Décimal
N7:17	PARECRIT_1_VALEUR	Décimal
N7:18	PARECRIT_2_ADRESSE	Décimal
N7:19	PARECRIT_2_VALEUR	Décimal
N7:20	PARECRIT_3_ADRESSE	Décimal
N7:21	PARECRIT_3_VALEUR	Décimal
N7:22	PARECRIT_4_ADRESSE	Décimal
N7:23	PARECRIT_4_VALEUR	Décimal
N7:24	PARECRIT_5_ADRESSE	Décimal

Adresse	Symbole	Affichage
N7:25	PARECRIT_5_VALEUR	Décimal
N7:26	PARECRIT_6_ADRESSE	Décimal
N7:27	PARECRIT_6_VALEUR	Décimal
N7:28	PARECRIT_7_ADRESSE	Décimal
N7:29	PARECRIT_7_VALEUR	Décimal
N7:30	PARECRIT_8_ADRESSE	Décimal
N7:31	PARECRIT_8_VALEUR	Décimal
O:1.13	WRPAR_SLAVE_FCT	Hexadécimal
O:1.14	WRPAR_ADRPAR	Hexadécimal
O:1.15	WRPAR_VALUE	Hexadécimal
I:1.13	CR_WRPAS_SLAVE_FCT	Hexadécimal
I:1.14	CR_WRPAS_ADRPAR	Hexadécimal
I:1.15	CR_WRPAS_VALUE	Hexadécimal
I:1.16	TRIGGER_IN_RD_WR	Hexadécimal
O:1.16	TRIGGER_OUT_RD_WR	Hexadécimal
N7:37	_____	Hexadécimal
B3:0/6	_____	Binaire

### Restrictions concernant l'exemple RSLogix 500

L'exemple présenté ici n'est pas parfait. Par exemple, en cas de réponse erronée (mauvais numéro d'esclave, de fonction, etc.), le programme n'effectue aucun traitement particulier et continue d'attendre une réponse jusqu'au timeout, alors que la passerelle n'a procédé à aucune ré-émission, puisque de son point de vue la réponse est correcte. En effet, l'ensemble du contenu de la réponse Modbus étant placé dans un champ « Data », il ne sera pas vérifié avant d'être recopié dans la mémoire de la passerelle. Seul le Checksum de la trame est vérifié par la passerelle.

Les deux « trigger bytes » situés dans le mot d'entrée I:1.16 ne sont pas utilisés. Vous devrez vous en servir si vous souhaitez que votre application soit informée de l'arrivée de chaque réponse associée aux deux commandes personnalisées « Transactions 1 » et « Transactions 2 ».

La compatibilité avec les différentes options offertes pour le champ « Control/Status Byte » de l'élément « ABC » (voir chapitre 5) n'est traitée que de manière partielle dans l'exemple présent. Les évolutions à apporter concernent principalement la gestion des bits 14 et 15 du mot de commande du maître DeviceNet et du mot d'état de la passerelle (bits 6 et 7 de l'entrée I:1.1 et de la sortie O:1.1 correspondantes). De plus, l'utilisation des diagnostics de la passerelle (champs EC et ED) reste à définir par l'utilisateur.

# Annexe D : Objets DeviceNet

## Présentation des objets DeviceNet de la passerelle

Le logiciel de la passerelle LUPP9 a été développé conformément à la **Modélisation des Objets** du protocole DeviceNet. De ce modèle découle une méthode d'adressage des données de la passerelle, appelées *Attributs*, constituée de quatre valeurs distinctes : ① l'Adresse du Nœud (MAC ID), ② l'Identificateur de la Classe de l'Objet (Class ID), ③ le Numéro de l'Instance (Instance ID) et ④ le Numéro de l'Attribut (Attribute ID). Une adresse constituée de cette manière est appelée un « **Chemin** ». La *Connexion par Messagerie Explicite*, par exemple, utilise de tels chemins pour échanger des données d'un point à l'autre sur un réseau DeviceNet.

Adresse	Min. – max.	Description
Nœud	0 – 00 063	Ce champ permet d'adresser un abonné parmi l'ensemble des abonnés d'un réseau DeviceNet grâce à son <b>MAC ID</b> .
Classe	1 – 65 535	Tous les objets partageant les mêmes caractéristiques appartiennent à une même classe, caractérisée par son <b>Class ID</b> .
Instance	0 – 65 535	Les instances représentent les différents objets d'une même classe. Toutes les instances d'une même classe partagent les mêmes comportements (1) et les mêmes <b>Attributs</b> , mais chacune d'entre elles possède son propre jeu de valeurs pour ces attributs. Lors de la création d'une instance (instanciation) par un abonné, celui-ci lui attribue un <b>Instance ID</b> unique, ce qui permet aux autres abonnés DeviceNet d'y avoir accès de manière individuelle.
Attribut	1 – 00 255	Chaque attribut représente l'une des caractéristiques des Instances appartenant à la même classe. Il se voit affecter une valeur de nature variable (octet, entier non signé, chaîne de caractères, etc.) dans le but de fournir des renseignements sur l'état de l'abonné ou pour effectuer des réglages sur les comportements (1) de l'abonné. <b>REMARQUE</b> : Pour accéder aux attributs de la Classe même d'un objet, il faut utiliser l'Instance 0x00 lors de la composition du Chemin complet. Exemple : Pour accéder à l'attribut « Revision » de la classe « Identity Object » de l'abonné DeviceNet n°4, le chemin à utiliser est le suivant : « 0x04 • 0x01 • 0x00 • 0x01 ».

- (1) Les comportements désignent les actions entreprises par un objet DeviceNet en réponse à des événements déterminés.

## Liste des objets DeviceNet de la passerelle

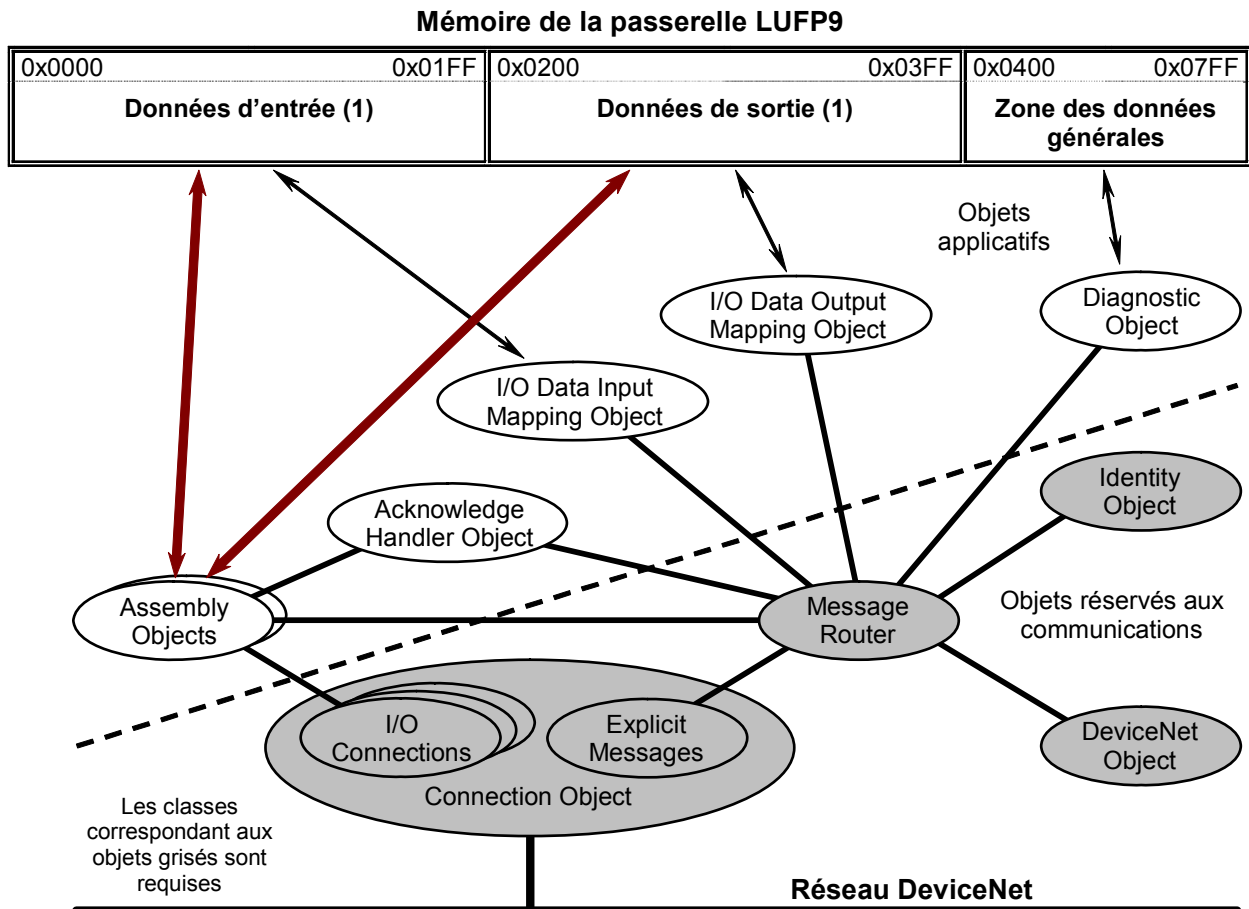
Classe	ID	Requis	Instances	Interfaces
Identity object	0x01	Oui	1	Message router
Message router	0x02	Oui	1	Explicit messages connection
DeviceNet object	0x03	Oui	1	Message router
Assembly object	0x04	Non	2 (1)	I/O connections ou Message router
Connection object	0x05	Oui	4 (2)	I/O connections ou Explicit messages
Acknowledge handler object	0x2B	Non	1	I/O connections ou Message router
I/O data input mapping object	0xA0	Non	1	Message router
I/O data output mapping object	0xA1	Non	1	Message router
Diagnostic object	0xAA	Non	1	Message router

- (1) Une zone d'entrée et une zone de sortie sont créées dans la mémoire de la passerelle.
- (2) Les quatre connexions instanciées sont : ① Explicit Connection, ② Polled Command/Response, ③ Bit Strobed Command/Response et ④ Change-of-State / Cyclic. Les trois dernières connexions sont du type « I/O Connection ».



# Annexe D : Objets DeviceNet

## Représentation graphique des objets DeviceNet de la passerelle



- (1) Les zones des données d'entrée et de sortie peuvent être lues ou écrites soit à l'aide des « I/O connections », soit à l'aide des « explicit messages ».

### Identity Object (classe 0x01)

L'objet « Identity » ne possède qu'une seule instance (Instance ID = 0x01). Cet objet contient des informations d'ordre général permettant d'identifier la passerelle et d'en diagnostiquer l'état. Cet objet est décrit dans le chapitre 6-2 du tome II des spécifications DeviceNet sur le site Web ODVA.

#### Attributs de la classe 0x01

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Requis	UINT	1	Indices majeur et mineur de la révision du « Identity Object ».

#### Services de la classe 0x01

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur de l'un des attributs de la classe.

# Annexe D : Objets DeviceNet

## Attributs de l'instance 0x01 de la classe 0x01

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get	<b>Vendor ID</b>	Requis	UINT	243
	L'ensemble des ID des fabricants de produits DeviceNet sont gérés par l'ODVA. Dans le cas de la passerelle LUF9, cet ID est égal à 243 (passerelles <i>Schneider Automation, Inc.</i> ).				
0x02	Get	<b>Device type</b>	Requis	UINT	12
	La liste des différents types de produits DeviceNet est gérée par l'ODVA. Cet attribut permet d'identifier le profil d'un abonné DeviceNet, et d'en déduire les exigences minimales et les options couramment utilisées par les abonnés de ce profil. La passerelle LUF9 correspond à un produit de type « Communication Adapter » (voir chapitre 3-7. du tome II des spécifications DeviceNet).				
0x03	Get	<b>Product code</b>	Requis	UINT	10937
	Cet attribut est géré par le fabricant du produit afin de caractériser ses propres produits. Il sert à identifier chacun de ses produits au sein de la même famille de produits (attribut « product type »). Cela permet de caractériser des produits présentant des différences au niveau de leur configuration et/ou de leurs options.				
0x04	Get	<b>Revision</b>	Requis	USINT , USINT	1, 36
	Indices de révision majeur et mineur permettant d'identifier le « Identity Object ». La valeur de chacun des deux membres de cet attribut ne peut pas être nulle. La représentation conventionnelle des indices de la révision est « majeur.mineur », avec 3 chiffres pour l'indice mineur, complétés à gauche par des zéros si besoin est. L'indice majeur est limité à 7 bits utiles. Son 8 <sup>ème</sup> bit est réservé et doit être égal à zéro.				
0x05	Get	<b>Status</b>	Requis	MOT	(registre de 16 bits)
	Cet attribut constitue un résumé de l'état général du produit. Il s'agit d'un registre de 16 bits : Bit 0 ..... Alloué à un maître (connexion maître/esclave prédéfinie). Bit 1 ..... Réserve (valeur = 2#0). Bit 2 ..... Produit configuré. Bits 3-7 ..... Réservés (valeur = 2#00000). Bit 8 ..... Faute mineure réparable. Bit 9 ..... Faute mineure irréversible. Bit 10 ..... Faute majeure réparable. Bit 11 ..... Faute majeure irréversible. Bits 12-15 ..... Réservés (valeur = 2#0000).				
0x06	Get	<b>Serial number</b>	Requis	UDINT	(variable)
	Le numéro de série du produit est combiné à l'attribut « vendor ID » afin de produire un identificateur unique pour chacun des produits DeviceNet. Chaque fabricant doit prendre la responsabilité de garantir que l'ensemble des produits DeviceNet qu'il fabrique dispose d'un numéro de série unique. Exemple de « numéro de série » : 0x 23 00 DD 20.				
0x07	Get	<b>Product name</b>	Requis	SHORT_STRING	« DeviceNet/Modbus Gateway »
	Cet attribut fournit une méthode d'identification visuelle et prend la forme d'une chaîne ASCII. Ce texte fournit une description courte du produit, ou de la famille de produits, équivalente à l'attribut « product code » (0x03). L'octet qui précède cette chaîne ASCII indique la longueur totale de cette chaîne, du premier au dernier caractère. Dans le cas de la passerelle LUF9, le nombre total d'octets compris dans l'attribut « product name » est égal à 24. La chaîne « DeviceNet/Modbus Gateway » comporte 24 caractères (espace compris). L'ensemble du contenu de l'attribut « product name », dans le cas de la passerelle LUF9, est donc égal à : 0x <b>18</b> 44 65 76 69 63 65 4E 65 74 2F 4D 6F 64 62 75 73 20 47 61 74 65 77 61 79. Les octets qui n'apparaissent pas en gras constituent le contenu de la chaîne ASCII (longueur = 0x18).				
0x09	Get	<b>Configuration consistency value</b>	Option	UINT	(variable)
	La valeur de cet attribut permet de vérifier la validité de la configuration du produit. Celui-ci modifie cet attribut de manière automatique lorsque la valeur d'un attribut non volatile est modifiée. Le comportement du produit lors de la détection d'une erreur d'intégrité de la configuration est spécifique à chaque type de produit. De même, la méthode de calcul de la valeur de cet attribut dépend entièrement du produit : CRC, compteur unitaire, etc. Cet attribut permet donc à un maître DeviceNet, par exemple, de vérifier que la configuration du produit DeviceNet n'a pas été modifiée. <b>REMARQUE</b> : En plus de calculer la valeur de cet attribut, la passerelle LUF9 utilise sa DEL <b>Ⓞ</b> GATEWAY pour signaler lorsque sa configuration n'est pas valide (DEL dans un état clignotant rouge/vert).				

## Services de l'instance 0x01 de la classe 0x01

Code du service	Nom du service	Exigence	Description
0x05	Reset	Requis	Ce service permet de réinitialiser la passerelle.
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur de l'un des attributs de l'instance.

# Annexe D : Objets DeviceNet

## Message Router Object (classe 0x02)

L'objet « Message Router » est l'élément par lequel tous les objets de type « Explicit messages » transitent afin d'être aiguillés vers les objets auxquels ils sont destinés. Il ne possède qu'une seule instance (Instance ID = 0x01). Cet objet est décrit dans le chapitre 6-3 du tome II des spécifications DeviceNet.

### Attributs de la classe 0x02

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Option	UINT	1	Indice de révision de la classe du « Message Router Object ».

### Services de la classe 0x02

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur de l'un des attributs de la classe.

### Attributs de l'instance 0x01 de la classe 0x02

Cette instance ne possède aucun attribut.

## DeviceNet Object (classe 0x03)

L'objet « DeviceNet » ne possède qu'une seule instance (Instance ID = 0x01). Cet objet contient l'état et la configuration générale du nœud de la passerelle sur le réseau DeviceNet. Il est décrit dans le chapitre 5-5 du tome II des spécifications DeviceNet. La passerelle LUF9 est un abonné du type « Group 2 only server » (voir chapitre 7-9 du tome I des spécifications DeviceNet).

### Attributs de la classe 0x03

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Requis	UINT	2	Indice de révision de la définition de la classe du « DeviceNet Object » actuellement utilisé pour l'implémentation des fonctions de communications DeviceNet de la passerelle. (1)

(1) Cet indice doit être compris entre 1 et 65 535 et sera incrémenté si la définition de la classe est remplacée par une définition plus récente.

### Services de la classe 0x03

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Option	Ce service permet de lire la valeur de l'un des attributs de la classe.

# Annexe D : Objets DeviceNet

## Attributs de l'instance 0x01 de la classe 0x03

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get	<b>MAC ID</b>	Requis	USINT	0 à 63
	La valeur de cet attribut correspond à l'adresse de la passerelle sur le réseau DeviceNet (MAC ID), c'est-à-dire à l'adresse qui est configurée à l'aide des commutateurs décrits dans le chapitre 2.7.2.				
0x02	Get	<b>Baud rate</b>	Option	USINT	0 à 2
	La valeur de cet attribut correspond à la vitesse de communication du réseau DeviceNet, telle qu'elle a été configurée sur la passerelle à l'aide des commutateurs décrits dans le chapitre 2.7.1. Cette vitesse doit être la même pour tous les abonnés du réseau DeviceNet. Les différentes valeurs possibles pour cet attribut sont : 0 (125 kbits/s), 1 (250 kbits/s) et 2 (500 kbits/s).				
0x03	Get/Set	<b>Bus-off Interrupt (BOI)</b>	Option	BOOL	0x00 ou 0x01
	Cet attribut est composé d'un seul bit qui définit comment la passerelle traite une interruption du bus. Avec la valeur par défaut 0x01, la passerelle réinitialise le contrôleur CAN et continue de communiquer toute interruption BOI détectée. Si la valeur est 0x00, la passerelle maintient la puce CAN en mode de désactivation du bus (réinitialisation) et passe en mode de défaut de communication lorsqu'une interruption BOI est détectée.				
0x04	Get/Set	<b>Bus Off Counter</b>	Option	USINT	0 à 255
	Nombre de fois que la puce CAN est passée en mode de désactivation du bus (comptabilisation du nombre d'interruptions BOI). Ce compteur est remis à zéro lors du démarrage ou de l'initialisation de l'équipement. Si plus de 255 interruptions BOI se produisent, la valeur du compteur reste 255. Le compteur n'est pas remis à zéro. Seul le service Set_Attribute_Single peut modifier cette valeur à ce stade ! La valeur d'un service Set_Attribute_Single est ignorée : le compteur est toujours remis à zéro.				
0x05	Get	<b>Allocation information</b>	Requis	BYTE , USINT	(variable)
	Cet attribut fournit des renseignements généraux sur la méthode d'allocation DeviceNet actuellement utilisée. Elle est composée du « choix d'allocation », au format BYTE, et du « MAC ID du maître », au format USINT et dont la valeur est comprise entre 0 et 63. Si le « MAC ID du maître » est égal à 255 (ce qui est le cas lors de l'initialisation de la passerelle), cela signifie qu'aucune allocation n'a eu lieu dans le cadre de l'utilisation du « Jeu Prédéfini des Connexions Maître/Esclave ». Reportez-vous aux chapitres 3-4., 5-5.4.2. et 7. du tome I des spécifications DeviceNet pour de plus amples détails à ce sujet. <i>Exemple</i> : 0x03, 0x00.				

## Services de l'instance 0x01 de la classe 0x03

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Option	Ce service permet de lire la valeur de l'un des attributs de l'instance.
0x4B	Allocate Master/Slave Connection Set	Option	Ce service permet d'allouer la connexion maître/esclave à un maître DeviceNet, sur demande de ce dernier.
0x4C	Release Master/Slave Connection Set	Option	Ce service permet de libérer la connexion maître/esclave précédemment allouée à un maître DeviceNet, sur demande de ce dernier.

# Annexe D : Objets DeviceNet

## Assembly Objects (Classe 0x04)

En règle générale, les objets de la classe « Assembly » servent à regrouper au sein d'un Attribut unique des Attributs (données) appartenant à des objets différents. Cela permet d'y avoir accès à l'aide d'un seul message. Dans le cas de la passerelle LUF9, cette classe possède seulement 2 instances, chacune d'entre elles étant affectée à la zone d'entrée (Instance ID = 0x64) ou à la zone de sortie (Instance ID = 0x96) de la passerelle. Cet objet est décrit dans le chapitre 6-5 du tome II des spécifications DeviceNet.

La première instance (Instance ID = 0x64) est affectée à la zone des données d'entrée de la passerelle. Cette zone d'entrée couvre l'ensemble des emplacements mémoire recevant une donnée issue d'une réponse Modbus à transmettre au maître DeviceNet. La seconde instance (Instance ID = 0x96) est affectée à la zone des données de sortie de la passerelle. Cette zone de sortie couvre l'ensemble des emplacements mémoire recevant une donnée à placer dans une requête Modbus, c'est-à-dire toutes les données qui sont transmises par le maître DeviceNet.

### Attributs de la classe 0x04

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Requis	UINT	2	Indice de révision de la classe du « Assembly Object ».
0x02	Get	Max instance	Option	UINT	0x96	Numéro maximum de chacune des instances créées au sein de la classe « Assembly Object ».

### Services de la classe 0x04

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Option	Ce service permet de lire la valeur de l'un des attributs de la classe.

### Attributs de l'instance 0x64 de la classe 0x04 (ENTREES MODBUS)

ID	Accès	Nom	Besoin	Type	Valeur
0x03	Get	<b>Data</b>	Requis	USINT [...]	(tableau de valeurs)
		Les données regroupées au sein de cet attribut correspondent à celles de l'attribut 0x01 de l'instance 0x01 de l'objet I/O Data Input Mapping. Dans le cas de la configuration par défaut, la taille de l'instance 0x64 (zone des données d'entrée de la passerelle) est égale à 32 octets et les données qui sont associées à l'attribut 0x03 de cette instance correspondent à ce qui est décrit dans Annexe B : Configuration par défaut, Zone mémoire des données d'entrée.			

### Attributs de l'instance 0x96 de la classe 0x04 (SORTIES MODBUS)

ID	Accès	Nom	Exigence	Type	Valeur
0x03	Get/Set	<b>Data</b>	Requis	USINT [...]	(tableau de valeurs)
		Les données regroupées au sein de cet attribut correspondent à celles de l'attribut 0x01 de l'instance 0x01 de l'objet I/O Data Output Mapping. Dans le cas de la configuration par défaut, la taille de l'instance 0x96 (zone des données de sortie de la passerelle) est égale à 32 octets et les données qui sont associées à l'attribut 0x03 de cette instance correspondent à ce qui est décrit dans Annexe B : Configuration par défaut, Zone mémoire des données de sortie.			

### Services des instances 0x64 et 0x96 de la classe 0x04

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire le tableau de valeurs qui correspond à l'attribut 0x03 de l'une des instances du « Assembly Object ».
0x10	Get_Attribute_Single	Option	Ce service permet d'écrire un tableau de valeurs dans le tableau de l'attribut 0x03 de l'une des instances du « Assembly Object ».

# Annexe D : Objets DeviceNet

## Connection Object (Classe 0x05)

Dans le cas de la passerelle LUF9, l'objet « Connection » possède jusqu'à quatre instances (Instance ID = 0x01 à 0x04). Chacune de ces instances représente l'une des deux extrémités d'une connexion virtuelle établie entre deux nœuds du réseau DeviceNet, en l'occurrence le nœud du maître DeviceNet et le nœud de la passerelle. Chaque instance de cet objet appartient à l'un des deux types de connexions suivants : Connexion explicite, qui permet de faire transiter des *Explicit Messages*, ou connexion implicite (*I/O Connections*). Cet objet est décrit dans le chapitre 5-4 du tome II des spécifications DeviceNet.

### AVERTISSEMENT

#### RISQUE DE FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

*Une seule connexion de type « I/O Connection » doit être utilisée à la fois.*

Etant donné que seules les zones « Input1 » et « Output1 » sont décrites dans ce guide et qu'il est interdit d'utiliser plus d'une « I/O connexion » dans la même zone, *vous ne devez pas configurer plus d'une « I/O connexion »*. Par exemple, si vous configurez une connexion « Polled », vous ne devez pas configurer une connexion « Strobed » ou « Change of State / Cyclic ».

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

Les quatre instances de l'objet « Connection » de la passerelle LUF9 sont brièvement décrites dans le tableau suivant, avant d'être détaillées ultérieurement de ce chapitre :

Instance ID	Type de connexion	Nom de la connexion
0x01	Explicit Messaging	Explicit Connection
0x02	I/O Connection	Polled Command/Response Connection <i>ou</i> Change-of-State / Cyclic consuming Connection
0x03	I/O Connection	Bit Strobed Command/Response Connection
0x04	I/O Connection	Change-of-State / Cyclic producing Connection

Chaque message d'une connexion de type « Explicit Messaging » contient l'adressage complet et les valeurs de l'Attribut concerné, ainsi que le Code de service décrivant l'action à entreprendre.

Chaque message d'une connexion de type « I/O Connection » contient uniquement des données d'E/S. L'ensemble des informations décrivant l'utilisation de ces données sont situées dans l'instance du « Connection Object » associée à ce message.

L'objet « Change-of-State / Cyclic Connection » (Instance ID 0x04) permet de sélectionner une connexion « Change-of-state » (COS) ou une connexion « Cyclic ».

La connexion « Change-of-state » permet à la passerelle de produire ses données uniquement sur changement de leurs valeurs ou sur expiration d'une temporisation appelée « heartbeat rate ». Une borne temporelle inférieure permet d'empêcher cette connexion de monopoliser la bande passante du réseau DeviceNet si les valeurs de ses données produites changent trop souvent.

Le passage en mode « Cyclic » permet de réduire les échanges effectués via cette connexion si la période de mise à jour (échantillonnage) des données produites est lente. En réglant le temps de cycle de la connexion sur la valeur de cette période, les données produites correspondent exactement aux échantillons des données, sans perte ni répétition d'échantillons.

### AVERTISSEMENT

#### FONCTIONNEMENT IMPREVU DU SYSTEME

Vous devez configurer l'objet « Change-of-State / Cyclic Connection » correctement. Dans le cas contraire, il affectera les communications sur l'ensemble du réseau DeviceNet network, provoquant la saturation du bus et l'absence de transmission de données depuis les autres esclaves.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

# Annexe D : Objets DeviceNet

## Attributs de la classe 0x05

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Option	UINT	1	Indice de révision de la classe du « Connection Object ».
0x64	Get/Set	Polled production	Option	USINT	0	Indice de la zone d'entrée utilisée par la passerelle pour les productions de sa connexion « Polled Command/Response ».
0x65	Get/Set	Polled consumption	Option	USINT	0	Indice de la zone de sortie utilisée par la passerelle pour les consommations de sa connexion « Polled Command/Response ».
0x66	Get/Set	Strobed production	Option	USINT	0	Indice de la zone d'entrée utilisée par la passerelle pour les productions de sa connexion « Bit Strobed Command/Response ».
0x67	Get/Set	Strobed consumption	Option	USINT	0	Indice de la zone de sortie utilisée par la passerelle pour les consommations de sa connexion « Bit Strobed Command/Response ».
0x68	Get/Set	COS production	Option	USINT	0	Indice de la zone d'entrée utilisée par la passerelle pour les productions de sa connexion « Bit Strobed Command/Response ».

## Services de la classe 0x05

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur de l'un des attributs de la classe.

## Attributs de l'instance 0x01 de la classe 0x05 : Explicit Connection

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get	<b>State</b>	Requis	USINT	0 à 5
Cet attribut représente l'état de l'objet « Explicit Connection ». Les valeurs supportées par la passerelle LUF9 sont les suivantes : 0 (inexistant), 1 (en cours de configuration), 3 (connexion établie), 4 (en timeout) et 5 (suppression différée). Reportez-vous aux figures 5.16 et 7.4 du tome I des spécifications DeviceNet pour de plus amples renseignements à ce sujet.					
0x02	Get	<b>Instance type</b>	Requis	USINT	0
Cet attribut définit le type de connexion de l'instance : Messaging connection (0) ou I/O connection (1).					
0x03	Get/Set	<b>Transport class trigger</b>	Requis	BYTE	0x83
Cet attribut définit le comportement de la connexion. Dans le cas de l'objet « Explicit Connection » de la passerelle LUF9, cet attribut prend la valeur 0x83, décomposée de la manière suivante : Bits 0-3 = 2#0011 .... Classe de transport = Classe 3. Bits 4-6 = 2#xxx..... Valeur ignorée dans le cas d'un serveur de données. Bits 7 = 2#1 ..... La passerelle se comporte comme un serveur de données répondant aux requêtes d'un client DeviceNet.					
0x04	Get/Set	<b>Produced connection ID</b>	Requis	UINT	2#11•••xx xxxx
La valeur de cet attribut est placée dans le Champ d'identification du protocole CAN lorsque la connexion passe en émission (messages du groupe 3). Le terme « xx xxxx » représente les 6 bits de l'adresse du nœud DeviceNet de la passerelle. Le terme « ••• » représente l'ID du message. Exemple : 0x070A = 2#111 0000 1010 (messages du groupe 3 ; ID des messages = 4 ; passerelle située à l'adresse 10).					
0x05	Get/Set	<b>Consumed connection ID</b>	Requis	UINT	2#11•••xx xxxx
La valeur de cet attribut correspond au contenu du Champ d'identification du protocole CAN des messages que la connexion doit recevoir (messages du groupe 3). Le terme « xx xxxx » représente les 6 bits de l'adresse du nœud DeviceNet. Le terme « ••• » représente l'ID du message. Exemple : 0x0601 = 2#110 0000 0001 (messages du groupe 3 ; ID des messages = 0 ; producteur situé à l'adresse 1).					

## Annexe D : Objets DeviceNet

ID	Accès	Nom	Besoin	Type	Valeur
0x06	Get/Set	<b>Initial comm. characteristics</b>	Requis	BYTE	0x21
		Cet attribut définit le ou les Groupes de Messages par lesquels les productions et les consommations associées à l'objet « Explicit Connection » sont effectuées. Reportez-vous aux chapitres 3-2. et 5-4.3.6. du tome I des spécifications DeviceNet pour de plus amples détails à ce sujet.			
0x07	Get/Set	<b>Produced connection size</b>	Requis	UINT	516
		Nombre maximum d'octets pouvant être transmis via la connexion de cette instance.			
0x08	Get/Set	<b>Consumed connection size</b>	Requis	UINT	516
		Nombre maximum d'octets pouvant être reçus via la connexion de cette instance.			
0x09	Get/Set	<b>Expected packet rate</b>	Requis	UINT	10 000 (unité = 1 ms, par pas de 10 ms)
		Cet attribut permet à la passerelle de calculer les valeurs de la <i>Temporisation du Déclenchement de l'Emission</i> et de la <i>Temporisation d'Inactivité / du Chien de Garde</i> pour les échanges effectués à l'aide de l'objet « Explicit Connection ». Reportez-vous au chapitre 5-4.4. du tome I des spécifications DeviceNet pour de plus amples renseignements au sujet de ces temporisations.			
0x0C	Get/Set	<b>Watchdog timeout action</b>	Requis	USINT	3
		Cet attribut définit l'action entreprise sur déclenchement du chien de garde ou sur inactivité de la connexion. Les différentes valeurs possibles sont les suivantes : 0 (Passage en timeout), 1 (Suppression Automatique) et 3 (Suppression Différée).			
0x0D	Get/Set	<b>Produced connection path length</b>	Requis	UINT	0
		Taille du tableau d'USINT de l'attribut 0x0E (chemin des productions de la connexion).			
0x0E	Get/Set	<b>Produced connection path</b>	Requis	USINT [...]	(chemin vide)
		Cet attribut définit le chemin local (sans MAC ID) de l'objet DeviceNet de la passerelle qui est utilisé pour produire les données de la connexion. Dans le cas présent, il n'y a pas de chemin de production pour la connexion « Explicit Connection ».			
0x0F	Get/Set	<b>Consumed connection path length</b>	Requis	UINT	0
		Taille du tableau d'USINT de l'attribut 0x10 (chemin des consommations de la connexion).			
0x10	Get/Set	<b>Consumed connection path</b>	Requis	USINT [...]	(chemin vide)
		Cet attribut définit le chemin local (sans MAC ID) de l'objet DeviceNet de la passerelle qui est destiné à recevoir les données consommées par la connexion. Dans le cas présent, il n'y a pas de chemin de consommation pour la connexion « Explicit Connection ».			
0x11	Get	<b>Production inhibit time</b>	Option	UINT	0
		Cet attribut définit le temps minimum, en millisecondes, entre de nouvelles productions de données. Une valeur nulle (valeur par défaut) indique l'absence de temps d'inhibition.			



# Annexe D : Objets DeviceNet

## Attributs de l'instance 0x02 de la classe 0x05 : Polled Command/Response Connection ou Change-of-State / Cyclic Consuming Connection

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get	<b>State</b>	Requis	USINT	0 à 4
	Cet attribut représente l'état de l'objet « Polled Command/Response Connection ». Les valeurs supportées par la passerelle LUF9 sont les suivantes : 0 (inexistant), 1 (en cours de configuration), 3 (connexion établie) et 4 (en timeout). Reportez-vous aux figures 5.16 et 7.4 du tome I des spécifications DeviceNet pour de plus amples renseignements à ce sujet.				
0x02	Get	<b>Instance type</b>	Requis	USINT	1
	Cet attribut définit le type de connexion de l'instance : Messaging connection (0) ou I/O connection (1).				
0x03	Get	<b>Transport class trigger</b>	Requis	BYTE	0x82
	Cet attribut définit le comportement de la connexion. Dans le cas de l'objet « Polled Command/Response Connection » de la passerelle LUF9, cet attribut prend la valeur 0x82, décomposée de la manière suivante : Bits 0-3 = 2#0010 .... Classe de transport = Classe 2. Bits 4-6 = 2#xxx..... Valeur ignorée dans le cas d'un serveur de données. Bits 7 = 2#1 ..... La passerelle se comporte comme un serveur de données répondant aux requêtes d'un client DeviceNet.				
0x04	Get/Set	<b>Produced connection ID</b>	Requis	UINT	2#0•• ••xx xxxx
	La valeur de cet attribut est placée dans le Champ d'identification du protocole CAN lorsque la connexion passe en émission (messages du groupe 1). Le terme « xx xxxx » représente les 6 bits de l'adresse du nœud DeviceNet de la passerelle. Le terme « •• •• » représente l'ID du message. Exemple : 0x03CA = 2#011 1100 1010 (messages du groupe 1 ; ID des messages = 15 ; passerelle située à l'adresse 10).				
0x05	Get/Set	<b>Consumed connection ID</b>	Requis	UINT	2#10x xxxx x•••
	La valeur de cet attribut correspond au contenu du Champ d'identification du protocole CAN des messages que la connexion doit recevoir (messages du groupe 2). Le terme « x xxxx x » représente les 6 bits de l'adresse du nœud DeviceNet. Le terme « ••• » représente l'ID du message. Exemple : 0x0455 = 2#100 0101 0101 (messages du groupe 2 ; ID des messages = 5 ; producteur situé à l'adresse 10).				
0x06	Get	<b>Initial comm. characteristics</b>	Requis	BYTE	0x01
	Cet attribut définit le ou les Groupes de Messages par lesquels les productions et les consommations associées à l'objet « Polled Command/Response Connection » sont effectuées. Reportez-vous aux chapitres 3-2. et 5-4.3.6. du tome I des spécifications DeviceNet pour de plus amples détails à ce sujet.				
0x07	Get	<b>Produced connection size</b>	Requis	UINT	(taille de la zone d'entrée)
	Nombre maximum d'octets pouvant être transmis via la connexion de cette instance. La valeur de cet attribut doit être égale à la taille de la zone d'entrée choisie à l'aide de l'attribut 0x0E. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut est égale à 32, c'est-à-dire à la taille de la zone d'entrée « Input1 ».				
0x08	Get	<b>Consumed connection size</b>	Requis	UINT	(taille de la zone de sortie)
	Nombre maximum d'octets pouvant être reçus via la connexion de cette instance. La valeur de cet attribut doit être égale à la taille de la zone de sortie choisie à l'aide de l'attribut 0x10. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut est égale à 32, c'est-à-dire à la taille de la zone « Output1 ».				
0x09	Get/Set	<b>Expected packet rate</b>	Requis	UINT	80 (unité = 1 ms, par pas de 10 ms)
	Cet attribut définit la périodicité des échanges effectués via les connexions de cette instance.				
0x0C	Get	<b>Watchdog timeout action</b>	Requis	USINT	0
	Cet attribut définit l'action entreprise sur déclenchement du chien de garde ou sur inactivité de la connexion. Les différentes valeurs possibles sont les suivantes : 0 (Passage en timeout), 1 (Suppression Automatique), 2 (RAZ Automatique) et 3 (Suppression Différée).				
0x0D	Get/Set	<b>Produced connection path length</b>	Requis	UINT	6
	Taille du tableau d'USINT de l'attribut 0x0E (chemin des productions de la connexion).				

# Annexe D : Objets DeviceNet

ID	Accès	Nom	Besoin	Type	Valeur
0x0E	Get/Set	<b>Produced connection path</b>	Requis	USINT [...]	0x 20 04 24 64 30 03
	<p>Cet attribut définit le chemin local (sans MAC ID) de l'objet DeviceNet de la passerelle qui est utilisé pour produire les données de la connexion. Dans le cas présent, le chemin de production par défaut pour la connexion « Polled Command/Response Connection » désigne l'attribut 0x03 de l'instance 0x64 de la classe 0x04, c'est-à-dire les données de la zone d'entrée « Input1 ».</p> <p><b>REMARQUE</b> : La modification de la valeur de l'attribut 0x64 de l'instance 0x00 de la classe 0x04 (paramètre EDS « Polled production ») a une influence directe sur la valeur de l'attribut présenté ici, puisque le chemin de la connexion correspondante est modifié pour qu'il permette d'avoir accès à la zone d'entrée sélectionnée. Ces modifications ne doivent avoir lieu qu'au moyen du fichier EDS fourni avec la passerelle.</p>				
0x0F	Get/Set	<b>Consumed connection path length</b>	Requis	UINT	6
	Taille du tableau d'USINT de l'attribut 0x10 (chemin des consommations de la connexion).				
0x10	Get/Set	<b>Consumed connection path</b>	Requis	USINT [...]	0x 20 04 24 96 30 03
	<p>Cet attribut définit le chemin local (sans MAC ID) de l'objet DeviceNet de la passerelle qui est destiné à recevoir les données consommées par la connexion. Dans le cas présent, le chemin de consommation par défaut pour la connexion « Polled Command/Response Connection » désigne l'attribut 0x03 de l'instance 0x96 de la classe 0x04, c'est-à-dire les données de la zone de sortie « Output1 ».</p> <p><b>REMARQUE</b> : La modification de la valeur de l'attribut 0x65 de l'instance 0x00 de la classe 0x04 (paramètre EDS « Polled consumption ») a une influence directe sur la valeur de l'attribut présenté ici, puisque le chemin de la connexion correspondante est modifié pour qu'il permette d'avoir accès à la zone de sortie sélectionnée. Ces modifications ne doivent avoir lieu qu'au moyen du fichier EDS fourni avec la passerelle.</p>				
0x11	Get	<b>Production inhibit time</b>	Requis	UINT	0
	Cet attribut définit le temps minimum, en millisecondes, entre de nouvelles productions de données. Une valeur nulle (valeur par défaut) indique l'absence de temps d'inhibition.				

## Attributs de l'instance 0x03 de la classe 0x05 : Bit Strobed Command/Response Connection

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get	<b>State</b>	Requis	USINT	0 à 4
	Cet attribut représente l'état de l'objet « Bit Strobed Command/Response Connection ». Les valeurs supportées par la passerelle LUF9 sont les suivantes : 0 (inexistant), 1 (en cours de configuration), 3 (connexion établie) et 4 (en timeout). Reportez-vous aux figures 5.16 et 7.4 du tome I des spécifications DeviceNet pour de plus amples renseignements à ce sujet.				
0x02	Get	<b>Instance type</b>	Requis	USINT	1
	Cet attribut définit le type de connexion de l'instance : Messaging connection (0) ou I/O connection (1).				
0x03	Get/Set	<b>Transport class trigger</b>	Requis	BYTE	0x83
	<p>Cet attribut définit le comportement de la connexion. Dans le cas de l'objet « Bit Strobed Command/Response Connection » de la passerelle LUF9, cet attribut prend la valeur 0x83, décomposée de la manière suivante :</p> <p>Bits 0-3 = 2#0011..... Classe de transport = Classe 3.</p> <p>Bits 4-6 = 2#xxx ..... Valeur ignorée dans le cas d'un serveur de données.</p> <p>Bits 7 = 2#1..... La passerelle se comporte comme un serveur de données répondant aux requêtes d'un client DeviceNet.</p>				
0x04	Get/Set	<b>Produced connection ID</b>	Requis	UINT	2#0●● ●●xx xxxx
	<p>La valeur de cet attribut est placée dans le Champ d'Identification du protocole CAN lorsque la connexion passe en émission (messages du groupe 1). Le terme « xx xxxx » représente les 6 bits de l'adresse du nœud DeviceNet de la passerelle. Le terme « ●● ●● » représente l'ID du message.</p> <p>Exemple : 0x038A = 2#011 1000 1010 (messages du groupe 1 ; ID des messages = 14 ; passerelle située à l'adresse 10).</p>				
0x05	Get/Set	<b>Consumed connection ID</b>	Requis	UINT	2#10x xxxx x●●●
	<p>La valeur de cet attribut correspond au contenu du Champ d'Identification du protocole CAN des messages que la connexion doit recevoir (messages du groupe 2). Le terme « x xxxx x » représente les 6 bits de l'adresse du nœud DeviceNet. Le terme « ●●● » représente l'ID du message.</p> <p>Exemple : 0x0400 = 2#100 0000 0000 (messages du groupe 2 ; ID des messages = 0 ; producteur situé à l'adresse 0).</p>				
0x06	Get/Set	<b>Initial comm. characteristics</b>	Requis	BYTE	0x02

# Annexe D : Objets DeviceNet

ID	Accès	Nom	Besoin	Type	Valeur
					Cet attribut définit le ou les Groupes de Messages par lesquels les productions et les consommations associées à l'objet « Bit Strobed Command/Response Connection » sont effectuées. Reportez-vous aux chapitres 3-2. et 5-4.3.6. du tome I des spécifications DeviceNet pour de plus amples détails à ce sujet.
0x07	Get/Set	<b>Produced connection size</b>	Requis	UINT	(taille de la zone d'entrée)
					Nombre maximum d'octets pouvant être transmis via la connexion de cette instance. La valeur de cet attribut doit être égale à la taille de la zone d'entrée choisie à l'aide de l'attribut 0x0E. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut est égale à 0, car aucune zone d'entrée n'est attribuée à l'objet « Bit Strobed Command/Response Connection ». <b>Taille maximale = 8 octets.</b>
0x08	Get/Set	<b>Consumed connection size</b>	Requis	UINT	(taille de la zone de sortie)
					La valeur de cet attribut n'est pas significative dans le cas de l'objet « Bit Strobed Command/Response Connection ». Cette valeur est fixée à 8.
0x09	Get/Set	<b>Expected packet rate</b>	Requis	UINT	80 (unité = 1 ms, par pas de 10 ms)
					Cet attribut définit la périodicité des échanges effectués via les connexions de cette instance.
0x0C	Get/Set	<b>Watchdog timeout action</b>	Requis	USINT	0
					Cet attribut définit l'action entreprise sur déclenchement du chien de garde ou sur inactivité de la connexion. Les différentes valeurs possibles sont les suivantes : 0 (Passage en timeout), 1 (Suppression Automatique), 2 (RAZ Automatique) et 3 (Suppression Différée).
0x0D	Get/Set	<b>Produced connection path length</b>	Requis	UINT	0
					Taille du tableau d'USINT de l'attribut 0x0E (chemin des productions de la connexion).
0x0E	Get/Set	<b>Produced connection path</b>	Requis	USINT [...]	(chemin de la zone)
					Cet attribut définit le chemin local (sans MAC ID) de l'objet DeviceNet de la passerelle qui est utilisé pour produire les données de la connexion. Dans le cas présent, le chemin de production pour la connexion « Bit Strobed Command/Response Connection » correspond à la zone d'entrée affectée à la connexion « Polled Command/Response Connection » à l'aide du paramètre EDS « Strobed production ».
0x0F	Get/Set	<b>Consumed connection path length</b>	Requis	UINT	0
					Taille du tableau d'USINT de l'attribut 0x10 (chemin des consommations de la connexion).
0x10	Get/Set	<b>Consumed connection path</b>	Requis	USINT [...]	(chemin de la zone)
					Cet attribut définit le chemin local (sans MAC ID) de l'objet DeviceNet de la passerelle qui est destiné à recevoir les données consommées par la connexion. Dans le cas présent, le chemin de consommation pour la connexion « Bit Strobed Command/Response Connection » correspond à la zone de sortie qui a été affectée à cette connexion à l'aide du paramètre EDS « Strobed consumption ».

## Attributs de l'instance 0x04 de la classe 0x05 : Change-of-State / Cyclic Producing Connection

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get	<b>State</b>	Requis	USINT	0 à 4
					Cet attribut représente l'état de l'objet « Change-of-State / Cyclic Producing Connection ». Les valeurs supportées par la passerelle LUF9 sont les suivantes : 0 (inexistant), 1 (en cours de configuration), 3 (connexion établie) et 4 (en timeout). Reportez-vous aux figures 5.16 et 7.4 du tome I des spécifications DeviceNet pour de plus amples renseignements à ce sujet.
0x02	Get	<b>Instance type</b>	Requis	USINT	1
					Cet attribut définit le type de connexion de l'instance : Messaging connection (0) ou I/O connection (1).
0x03	Get/Set	<b>Transport class trigger</b>	Requis	BYTE	0x12 ou 0x02
					Cet attribut définit le comportement de la connexion. Dans le cas de l'objet « Change-of-State / Cyclic Producing Connection » de la passerelle LUF9, cet attribut prend la valeur 0x12 ou 0x02, décomposée de la manière suivante : Bits 0-3 = 2#0010 ..... Classe de transport = Classe 2. Bits 4-6 = 2#001 ou 2#000 ..... Mode « Change-of-State » (2#001) ou mode « Cyclic » (2#000). Bits 7 = 2#0

# Annexe D : Objets DeviceNet

ID	Accès	Nom	Besoin	Type	Valeur
0x04	Get/Set	<b>Produced connection ID</b>	Requis	UINT	2#0●● ●●xx xxxx
		La valeur de cet attribut est placée dans le Champ d'identification du protocole CAN lorsque la connexion passe en émission (messages du groupe 1). Le terme « xx xxxx » représente les 6 bits de l'adresse du nœud DeviceNet de la passerelle. Le terme « ●● ●● » représente l'ID du message. Exemple : 0x034A = 2#011 0100 1010 (messages du groupe 1 ; ID des messages = 13 ; passerelle située à l'adresse 10).			
0x05	Get/Set	<b>Consumed connection ID</b>	Requis	UINT	2#10x xxxx x●●
		La valeur de cet attribut correspond au contenu du Champ d'identification du protocole CAN des messages que la connexion doit recevoir (messages du groupe 2). Le terme « x xxxx x » représente les 6 bits de l'adresse du nœud DeviceNet. Le terme « ●● » représente l'ID du message. Exemple : 0x0452 = 2#100 0101 0010 (messages du groupe 2 ; ID des messages = 2 ; passerelle située à l'adresse 10).			
0x06	Get/Set	<b>Initial comm. characteristics</b>	Requis	BYTE	0x01
		Cet attribut définit le ou les Groupes de Messages par lesquels les productions et les consommations associées à l'objet « Change-of-State / Cyclic Producing Connection » sont effectuées. Dans le cas présent, il désigne les groupes 1 et 2. Reportez-vous aux chapitres 3-2. et 5-4.3.6. du tome I des spécifications DeviceNet pour de plus amples détails à ce sujet.			
0x07	Get/Set	<b>Produced connection size</b>	Requis	UINT	(taille de la zone d'entrée)
		Nombre maximum d'octets pouvant être transmis via la connexion de cette instance. La valeur de cet attribut doit être égale à la taille de la zone d'entrée choisie à l'aide de l'attribut 0x0E. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut est égale à 0, car aucune zone d'entrée n'est attribuée à l'objet « Change-of-State / Cyclic Producing Connection ».			
0x08	Get/Set	<b>Consumed connection size</b>	Requis	UINT	0
		Nombre maximum d'octets pouvant être reçus via la connexion de cette instance. Puisque la passerelle LUF9 ne consomme aucune donnée via cette connexion, la valeur de cet attribut restera égale à 0.			
0x09	Get/Set	<b>Expected packet rate</b>	Requis	UINT	0 (unité = 1 ms, par pas de 10 ms)
		Cet attribut définit la périodicité des échanges effectués via les connexions de cette instance.			
0x0C	Get/Set	<b>Watchdog timeout action</b>	Requis	USINT	0
		Cet attribut définit l'action entreprise sur déclenchement du chien de garde ou sur inactivité de la connexion. Les différentes valeurs possibles sont les suivantes : 0 (Passage en timeout), 1 (Suppression Automatique), 2 (RAZ Automatique) et 3 (Suppression Différée).			
0x0D	Get/Set	<b>Produced connection path length</b>	Requis	UINT	0
		Taille du tableau d'USINT de l'attribut 0x0E (chemin des productions de la connexion).			
0x0E	Get/Set	<b>Produced connection path</b>	Requis	USINT [...]	(chemin de la zone)
		Cet attribut définit le chemin local (sans MAC ID) de l'objet DeviceNet de la passerelle qui est utilisé pour produire les données de la connexion. Dans le cas présent, le chemin de production pour la connexion « Change-of-State / Cyclic Producing Connection » correspond à la zone de sortie qui a été affectée à cette connexion à l'aide du paramètre EDS « COS production ».			
0x0F	Get/Set	<b>Consumed connection path length</b>	Requis	UINT	4
		Taille du tableau d'USINT de l'attribut 0x10 (chemin des consommations de la connexion).			
0x10	Get/Set	<b>Consumed connection path</b>	Requis	USINT [...]	(chemin de la zone)
		Cet attribut définit le chemin local (sans MAC ID) de l'objet DeviceNet de la passerelle qui est destiné à recevoir les données consommées par la connexion. Dans le cas présent, le chemin de consommation pour la connexion « Change-of-State / Cyclic Producing Connection » désigne l'instance 0x01 de la classe 0x2B, c'est-à-dire l'unique objet de la classe « Acknowledge Handler Object ». <b>REMARQUE</b> : Le fichier EDS fourni avec la passerelle ne contient pas de paramètre dont la modification ait une quelconque influence sur la valeur de cet attribut.			

# Annexe D : Objets DeviceNet

## Attributs des instances 0x01 à 0x04 de la classe 0x05

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur d'un attribut de l'une des instances du « Connection Object ».
0x10	Set_Attribute_Single	Option	Ce service permet d'écrire la valeur d'un attribut de l'une des instances du « Connection Object ».

## Acknowledge Handler Object (classe 0x2B)

L'objet « Acknowledge Handler » ne possède qu'une seule instance (Instance ID = 0x01). Cet objet est utilisé par les connexions dont le producteur a besoin de savoir si ses données ont été reçues par son, ou ses, destinataires (consommateurs). Cet objet est décrit dans le chapitre 6-31 du tome II des spécifications DeviceNet.

## Attributs de la classe 0x2B

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Option	UINT	1	Indice de révision de la classe du « Acknowledge Handler Object ».
0x02	Get	Max instance	Option	UINT	1	Numéro maximum de chacune des instances créées au sein de la classe « Acknowledge Handler Object ».

## Services de la classe 0x2B

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur de l'un des attributs de la classe.

## Attributs de l'instance 0x01 de la classe 0x2B

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get/Set	<b>Acknowledge timer</b>	Requis	UINT	20 (unité : 1ms)
	La valeur de cet attribut détermine la durée de l'attente de l'acquiescement du message d'une connexion. Une fois cette durée écoulée, la passerelle procède à la ré-émission du message qui vient de ne pas être acquitté. La valeur de cet attribut doit être comprise entre 1 et 65 535, et sa valeur par défaut est égale à 20.				
0x02	Get/Set	<b>Retry limit</b>	Requis	USINT	1
	Cet attribut détermine le nombre maximum autorisé de déclenchements successifs du timeout d'acquiescement pour un même message, et donc le nombre de ré-émissions autorisé pour chaque message. La valeur de cet attribut doit être comprise entre 0 et 255, et sa valeur par défaut est égale à 1.				
0x03	Get/Set	<b>COS producing connection instance</b>	Requis	UINT	4
	La valeur de cet attribut est égale au numéro de l'instance (Instance ID) de la classe « Connection Object » qui correspond à la connexion de type « Change-of-State » associée à l'objet « Acknowledge Handler ». Cette association permet à ce dernier de transmettre les acquiescements qu'il reçoit à la connexion correspondante s'ils sont destinés à celle-ci.				
0x04	Get	<b>Ack list size</b>	Option	BYTE	1
	Cet attribut représente le nombre maximum de membres qu'il est possible de placer dans la liste des acquiescements. Si la valeur de cet attribut est nulle, la taille de la liste est dynamique, ce qui n'est pas le cas de la passerelle LUFFP9.				
0x05	Get	<b>Ack list</b>	Option	BYTE , USINT [...]	0 , (liste vide)
	Cet attribut correspond à la liste des instances actives de la classe « Connection Object » pour lesquelles la réception d'un acquiescement est nécessaire. Il est composé de deux éléments : le nombre de membres (BYTE) et la liste des numéros des instances de la classe « Connection Object » associées (USINT [...]). La taille de la liste est égale à la valeur du premier élément. Par défaut, la liste est vide (absence du terme de type USINT [...]) et seul l'élément BYTE est créé.  Exemple : « 1, 4 » pour une liste ne comportant qu'une seule instance de la classe « Connection Object ». Cette instance (0x04) correspond à la connexion « Change-of-State /Cyclic Producing Connection ».				

## Annexe D : Objets DeviceNet

ID	Accès	Nom	Besoin	Type	Valeur
0x06	Get	<b>Data with ack path list size</b>	Option	BYTE	1
Cet attribut représente le nombre maximum de membres qu'il est possible de placer dans la liste des chemins des données d'acquiesement. Si la valeur de cet attribut est nulle, la taille de la liste est dynamique, ce qui n'est pas le cas de la passerelle LUF9.					
0x07	Get	<b>Data with ack path list</b>	Option	BYTE, (UINT, USINT, USINT[...])[...]	(liste des chemins des données d'acquiesement)
Cet attribut correspond à la liste des associations « instance de connexion / objet applicatif consommateur » permettant de rediriger les données reçues dans un acquiesement. Un acquiesement ne contient pas forcément des données et cet attribut est donc optionnel. Celui-ci est composé des éléments suivants : <ul style="list-style-type: none"> <li>• Le nombre de membres de la liste (BYTE).</li> <li>• La liste des associations « instance de connexion / objet applicatif consommateur » ( UINT , USINT, USINT [...] ) [...]. La taille de cette liste est égale à la valeur du premier élément, décrit ci-dessus, et elle est elle-même composée des éléments suivants : <ul style="list-style-type: none"> <li>- Le numéro de l'instance de la connexion COS acquiesée (UINT).</li> <li>- La longueur du chemin de l'objet DeviceNet destiné à recevoir les données de l'acquiesement (USINT).</li> <li>- Le chemin de l'objet DeviceNet destiné à recevoir les données de l'acquiesement (USINT [...])</li> </ul> </li> </ul> Exemple : 0x 01 64 00 06 20 04 24 64 30 03. La valeur de cet attribut signifie que cette liste ne contient qu'un seul élément (0x01), celui-ci faisant référence à l'instance 0x0064 et que le chemin des données d'acquiesement (0x06 : longueur de 6 octets) fait référence à l'attribut 0x03 de l'instance 0x64 de la classe 0x04, c'est-à-dire aux données d'entrée Modbus.					

### Services de l'instance 0x01 de la classe 0x2B

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur d'un attribut de l'unique instance du « Acknowledge Handler Object ».
0x10	Set_Attribute_Single	Requis	Ce service permet d'écrire la valeur d'un attribut de l'unique instance du « Acknowledge Handler Object ».

# Annexe D : Objets DeviceNet

## I/O Data Input Mapping Object (classe 0xA0)

L'objet « I/O Data Input Mapping Object » ne possède qu'une seule instance (Instance ID = 0x01) et est spécifique à la passerelle LUF9. Il contient l'ensemble des données de l'unique zone d'entrée de la passerelle. L'unique attribut (Attribute ID = 0x01) de l'instance de cet objet est associé à la zone d'entrée « Input1 ». Cette zone d'entrée couvre l'ensemble des emplacements mémoire recevant une donnée issue d'une réponse Modbus.

### Attributs de la classe 0xA0

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Option	UINT	1	Indice de révision de la classe du « I/O Data Input Mapping Object ».
0x64	Get/Set	Input1 offset	Option	USINT	0x0000	Adresse relative du début de la zone d'entrée n°1. (1)
0x6E	Get/Set	Input1 length	Option	USINT	0x0020	Taille, exprimée en octets, de la zone d'entrée n°1. (1)

- (1) Ces 2 attributs correspondent aux paramètres « Param6 » et « Param7 » référencés par le fichier EDS fourni avec la passerelle. Leur accès en écriture (Accès = Set) est réservé aux outils de configuration DeviceNet, puisqu'il permet de modifier l'emplacement ou la taille de cette zone de données d'entrée. Le service « Set\_Attribute\_Single » ne devra donc pas être utilisé avec ces attributs. La modification de l'un de ces deux attributs a des conséquences directes sur l'attribut 0x01 de l'instance 0x01 de l'objet « I/O Data Input Mapping Object » (taille des données). Cet attribut n'est pas créé si la taille de la zone d'entrée de la passerelle est nulle. L'attribut « Input1 offset » correspond à un décalage depuis le début de la zone mémoire réservée aux données d'entrée (0x0000).

Les valeurs situées dans la colonne « Valeur » correspondent à la configuration par défaut de la passerelle LUF9 (zone « Input1 » située à l'adresse 0x0000 et comportant 32 octets).

### Services de la classe 0xA0

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur de l'un des attributs de la classe.

### Attributs de l'instance 0x01 de la classe 0xA0

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get	<b>Data</b>	Option	USINT [...]	(zone d'entrée n°1)
Cet attribut correspond à la zone d'entrée « Input1 » de la passerelle. Sa lecture permet d'obtenir les valeurs de l'ensemble des données contenues dans cette zone sous la forme d'un tableau d'octets dont la taille correspond à celle de la zone. Ce même attribut est également impliqué dans l'utilisation de l'instance Assembly Objects décrite dans Annexe D : Objets DeviceNet.					
<b>REMARQUE</b> : Dans le cas de la configuration par défaut, l'attribut 0x01 correspond à un tableau de 32 octets dont le contenu est décrit dans Annexe B : , Zone mémoire des données d'entrée.					

### Services de l'instance 0x01 de la classe 0xA0

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire le tableau de valeurs du seul attribut de l'unique instance du « I/O Data Input Mapping Object ».

# Annexe D : Objets DeviceNet

## I/O Data Output Mapping Object (classe 0xA1)

L'objet « I/O Data Output Mapping Object » ne possède qu'une seule instance (Instance ID = 0x01) et est spécifique à la passerelle LUF9. Il contient l'ensemble des données de l'unique zone de sortie de la passerelle. L'unique attribut (Attribute ID = 0x01) de l'instance de cet objet est associé à la zone de sortie « Output1 ». Cette zone de sortie couvre l'ensemble des emplacements mémoire dont les valeurs sont transmises aux esclaves Modbus via des requêtes Modbus.

### Attributs de la classe 0xA1

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Option	UINT	1	Indice de révision de la classe du « I/O Data Output Mapping Object ».
0x64	Get/Set	Output1 offset	Option	USINT	0x0000	Adresse relative du début de la zone de sortie n°1. (1)
0x6E	Get/Set	Output1 length	Option	USINT	0x0020	Taille, exprimée en octets, de la zone de sortie n°1. (1)

- (1) Ces 2 attributs correspondent aux paramètres « Param18 » et « Param19 » référencés par le fichier EDS fourni avec la passerelle. Leur accès en écriture (Accès = Set) est réservé aux outils de configuration DeviceNet, puisqu'il permet de modifier l'emplacement ou la taille de cette zone de données de sortie. Le service « Set\_Attribute\_Single » ne devra donc pas être utilisé avec ces attributs. La modification de l'un de ces deux attributs a des conséquences directes sur l'attribut 0x01 de l'instance 0x01 de l'objet « I/O Data Output Mapping Object » (taille des données). Cet attribut n'est pas créé si la taille de la zone de sortie de la passerelle est nulle. L'attribut « Output1 offset » correspond à un décalage depuis le début de la zone mémoire réservée aux données de sortie (0x0200).

Les valeurs situées dans la colonne « Valeur » correspondent à la configuration par défaut de la passerelle LUF9 (zone « Output1 » située à l'adresse 0x0200 et comportant 32 octets).

### Services de la classe 0xA1

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur de l'un des attributs de la classe.

### Attributs de l'instance 0x01 de la classe 0xA1

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get/Set	Data	Option	USINT [...]	(zone de sortie n°1)
Cet attribut correspond à la zone de sortie « Output1 » de la passerelle. Sa lecture permet d'obtenir les valeurs de l'ensemble des données contenues dans cette zone, et son écriture permet de les modifier. Ces valeurs prennent la forme d'un tableau d'octets dont la taille correspond à celle de la zone. Ce même attribut est également impliqué dans l'utilisation de l'instance 0x96 de l'objet « Assembly Objects » décrite dans Annexe D : Objets DeviceNet.					
<b>REMARQUE</b> : Dans le cas de la configuration par défaut, l'attribut 0x01 correspond à un tableau de 32 octets dont le contenu est décrit dans Annexe B : , Zone mémoire des données de sortie.					

### Services de l'instance 0x01 de la classe 0xA1

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Option	Ce service permet de lire le tableau de valeurs du seul attribut de l'unique instance du « I/O Data Output Mapping Object ».
0x10	Set_Attribute_Single	Requis	Ce service permet d'écrire / de modifier toutes les valeurs du seul attribut de l'unique instance du « I/O Data Output Mapping Object ».



# Annexe D : Objets DeviceNet

## Diagnostic Object (Classe 0xAA)

L'objet « Diagnostic Object » ne possède qu'une seule instance (Instance ID = 0x01) et est spécifique à la passerelle LUF9. Il contient un grand nombre de données de diagnostic de tous niveaux. Par conséquent, certaines d'entre elles ne devront pas être utilisées dans le cadre de la mise en œuvre de la passerelle, car elles sont réservées aux opérations de maintenance effectuées sur la passerelle ou lors du développement de son logiciel. Cependant, les attributs auxquels elles correspondent sont tous décrits ci-dessous dans un souci d'exhaustivité.

### Attributs de la classe 0xAA

ID	Accès	Nom	Besoin	Type	Valeur	Description
0x01	Get	Revision	Option	UINT	1	Indice de révision de la classe de l'objet « Diagnostic Object ».

### Services de la classe 0xAA

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur de l'un des attributs de la classe.

### Attributs de l'instance 0x01 de la classe 0xAA

ID	Accès	Nom	Besoin	Type	Valeur
0x01	Get	<b>DeviceNet module serial number</b>	Option	UDINT	(variable)
		La valeur de « DeviceNet module serial number » correspond au numéro de série de la carte <i>AnyBus-S DeviceNet</i> de la passerelle, c'est-à-dire la carte sur laquelle sont situés le bloc de commutateurs et le connecteur DeviceNet. Exemple : 0x 20 DD 00 23.			
0x02	Get	<b>Vendor ID</b>	Option	UINT	0x0001
		La valeur de cet attribut est égale à 0X0001 dans le cas de la passerelle LUF9. Il est impossible d'utiliser la valeur 0x0000 et les valeurs comprises entre 0x0002 et 0xFFFF sont réservées aux fournisseurs de la passerelle.			
0x03	Get	<b>Fieldbus type</b>	Option	UINT	0x0025
		Dans le cas de la passerelle LUF9, cet attribut prend toujours la même valeur (0x0025), puisqu'il caractérise le réseau DeviceNet. Toute autre valeur serait erronée (exemple : 0x0001 pour un réseau Profibus-DP).			
0x04	Get	<b>DeviceNet module software version</b>	Option	UINT	0x0136
		Cet attribut indique la version du logiciel présent dans la carte <i>AnyBus-S DeviceNet</i> de la passerelle. L'indice majeur de cette version est donné par l'octet de poids fort et son indice mineur est donné par l'octet de poids faible, tous deux au format BCD. Exemple : 0x0136 correspond à la version 01.36.			
0x05	Get	<b>Interrupt count</b>	Option	UINT	(compteur)
		La valeur du « compteur d'interruptions » est incrémentée d'une unité à chaque fois qu'une interruption liée à la gestion du réseau aval Modbus se produit.			
0x06	Get	<b>Watchdog counter in</b>	Option	UINT	0x0000
		Ce compteur n'est pas implémenté. L'utilisation de cet attribut est donc inutile. La fonction première de ce compteur consiste à assurer une <i>feedback</i> du compteur de vie représenté par l'attribut 0x07, ce qui permettrait à la carte <i>AnyBus-S DeviceNet</i> de s'assurer que la carte à laquelle elle est connectée fonctionne correctement en comparant les valeurs de ces deux attributs.			
0x07	Get	<b>Watchdog counter out</b>	Option	UINT	(compteur)
		La valeur de ce compteur est incrémentée d'une unité toutes les millisecondes (au minimum une écriture toutes les 50 ms) et fait fonction de compteur de vie interne, à l'attention de la carte applicative de la passerelle, c'est-à-dire la carte à laquelle se superpose la carte <i>AnyBus-S DeviceNet</i> .			

# Annexe D : Objets DeviceNet

ID	Accès	Nom	Besoin	Type	Valeur
0x09	Get	<b>LED status</b>	Option	USINT [6]	(variable)
	Les valeurs des éléments de cet attribut correspondent à l'état des 6 DEL de la passerelle (1 octet par DEL). Le premier octet correspond à la DEL ①, le deuxième à la DEL ②, etc., jusqu'à la DEL ⑥. Chaque octet prend l'une des valeurs suivantes pour désigner l'état de la DEL à laquelle il correspond : 0x00 (DEL éteinte), 0x01 (DEL verte) ou 0x02 (DEL rouge).				
0x0A	Get	<b>Module type</b>	Option	UINT	0x0101
	La valeur de cet attribut est toujours égale à 0x0101 dans le cas de la passerelle LUF9, car il s'agit d'un module de type « <i>AnyBus-S</i> ».				
0x0B	Get	<b>DeviceNet module status</b>	Option	USINT	(registre de 8 bits)
	La lecture des bits de cet attribut permet de connaître certains renseignements sur l'état de la carte <i>AnyBus-S DeviceNet</i> de la passerelle. Les quatre bits utiles de ces registres sont décrits ci-dessous : Bit 0 : Passerelle hors ligne (0) / en ligne (1) sur le réseau DeviceNet. Bit 1 : Toutes les sorties sont remises à zéro (0) ou maintenues (1) dans la zone mémoire de sortie si la passerelle est hors ligne sur le réseau DeviceNet. Bit 8 : Toutes les entrées sont remises à zéro (0) ou maintenues (1) dans la zone mémoire d'entrée si l'application de la passerelle est arrêtée. Bit 9 : Registre d'indication de changement d'état des sorties désactivé (0) / activé (1).				
0x0C	Get	<b>Changed data field</b>	Option	LWORD	—
	Chacun des bits de ce registre de 64 bits indique si le contenu de 8 octets consécutifs de la zone mémoire de sortie a été modifié. Le bit 0 concerne les octets 0x0200 à 0x0207, le bit 1 concerne les octets 0x0208 à 0x0215, etc., jusqu'au bit 63, qui concerne les octets 0x03F8 to 0x03FF.				
0x0D	Get	<b>Interrupt cause</b>	Option	BYTE	(registre de 8 bits)
	Ce registre permet de déterminer la cause de la dernière interruption survenue. Chaque bit est activé lorsque l'événement associé se produit, puis il est remis à zéro par le gestionnaire d'interruption de la passerelle. Ce registre n'est donc pas destiné à être utilisé par le maître DeviceNet. Bit 0 : Passage de la passerelle en ligne sur le réseau DeviceNet. Bit 1 : Passage de la passerelle hors ligne sur le réseau DeviceNet. Bit 2 : Modification des données.				
0x0E	Get	<b>Interrupt notification</b>	Option	BYTE	(registre de 8 bits)
	Ce registre permet de déterminer quels types d'interruptions sont autorisés (voir description de l'attribut 0x0D). Sa valeur est fixée lors de l'initialisation de la passerelle, au moyen d'un télégramme spécifique (non décrit dans ce guide). Bit 0 : Génération d'une interruption lors du passage en ligne de la passerelle sur le réseau DeviceNet. Bit 1 : Génération d'une interruption lors du passage hors ligne de la passerelle sur le réseau DeviceNet. Bit 2 : Génération d'une interruption lors d'une modification des données. Pour cela, le registre de changement d'état des sorties doit être activé (voir description du bit 9 de l'attribut 0x0B).				
0x0F	Get	<b>IN cyclic I/O length</b>	Option	UINT	0x0020
	Cet attribut indique la taille totale des données d'entrée cycliques ( <i>I/O IN data</i> ), exprimée en nombre d'octets. Cette taille couvre l'ensemble de l'espace mémoire de la passerelle occupé par les données d'entrée Modbus, les emplacements libres étant également comptabilisés. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut correspond à la taille de la zone d'entrée de la passerelle, c'est-à-dire 32 octets.				
0x10	Get	<b>IN DPRAM length</b>	Option	UINT	0x0020
	Cet attribut indique la taille totale des données et des paramètres d'entrée dans la mémoire de la passerelle ( <i>valid IN bytes in DPRAM</i> ), exprimée en nombre d'octets. Cette taille couvre l'ensemble de l'espace mémoire de la passerelle occupé par les données et les paramètres d'entrée Modbus, les emplacements libres étant également comptabilisés. Puisque aucun paramètre d'entrée n'est défini, les valeurs des attributs 0x0F et 0x10 sont toutes deux identiques. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut est égale à 32 octets.				

## Annexe D : Objets DeviceNet

ID	Accès	Nom	Besoin	Type	Valeur
0x11	Get	<b>IN total length</b>	Option	UINT	0x0020
<p>Cet attribut indique la taille totale des données d'entrée utilisées dans la mémoire interne étendue de la passerelle (<i>IN bytes supported</i>), exprimée en nombre d'octets. Cette taille est égale à la valeur de l'attribut précédent (taille des entrées en DPRAM), car elle ne contient que des données d'entrée. Les valeurs des attributs 0x0F, 0x10 et 0x11 sont toutes identiques. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut est égale à 32 octets.</p> <p><b>REMARQUE</b> : La mémoire interne étendue de la passerelle est différente de la mémoire DPRAM, dont il est question dans le reste du présent guide. Par conséquent, vous n'aurez pas à vous en soucier dans le cadre d'une utilisation normale de la passerelle.</p>					
0x12	Get	<b>OUT cyclic I/O length</b>	Option	UINT	0x0020
<p>Cet attribut indique la taille totale des données de sortie cycliques (<i>I/O OUT data</i>), exprimée en nombre d'octets. Cette taille couvre l'ensemble de l'espace mémoire de la passerelle occupé par les données de sortie Modbus, les emplacements libres étant également comptabilisés. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut correspond à la taille de la zone de sortie de la passerelle, c'est-à-dire 32 octets.</p>					
0x13	Get	<b>OUT DPRAM length</b>	Option	UINT	0x0020
<p>Cet attribut indique la taille totale des données et des paramètres de sortie dans la mémoire de la passerelle (<i>valid OUT bytes in DPRAM</i>), exprimée en nombre d'octets. Cette taille couvre l'ensemble de l'espace mémoire de la passerelle occupé par les données et les paramètres de sortie Modbus, les emplacements libres étant également comptabilisés. Puisque aucun paramètre de sortie n'est défini, les valeurs des attributs 0x12 et 0x13 sont toutes deux identiques. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut est égale à 32 octets.</p>					
0x14	Get	<b>OUT total length</b>	Option	UINT	0x0020
<p>Cet attribut indique la taille totale des données de sortie utilisées dans la mémoire interne étendue de la passerelle (<i>OUT bytes supported</i>), exprimée en nombre d'octets. Cette taille est égale à la valeur de l'attribut précédent (taille des sorties en DPRAM), car elle ne contient que des données de sortie. Les valeurs des attributs 0x12, 0x13 et 0x14 sont toutes identiques. Dans le cas de la configuration par défaut de la passerelle LUF9, la valeur de cet attribut est égale à 32 octets.</p> <p><b>REMARQUE</b> : La mémoire interne étendue de la passerelle est différente de la mémoire DPRAM, dont il est question dans le reste du présent guide. Par conséquent, vous n'aurez pas à vous en soucier dans le cadre d'une utilisation normale de la passerelle.</p>					

### Services de l'instance 0x01 de la classe 0xAA

Code du service	Nom du service	Besoin	Description
0x0E	Get_Attribute_Single	Requis	Ce service permet de lire la valeur d'un attribut de l'unique instance du « Diagnostic Object ».

# Annexe E : Commandes Modbus

Les seules commandes Modbus autorisées par la passerelle sont présentées ci-contre. La structure des trames de la requête et de la réponse de chacune d'entre elles est ensuite décrite dans les chapitres suivants.

Code fonction		Diffusion (1)	Commande Modbus
03	0x03	—	Read Holding Registers
06	0x06	Oui	Preset Single Register
16	0x10	Oui	Preset Multiple Registers

(1) Cette colonne indique si la commande peut être ajoutée (« Oui ») ou non (« — ») dans la liste des commandes d'un nœud de diffusion, appelé « Broadcaster » dans ABC-LUFP Config Tool.

Dans les chapitres suivants, chacun des octets des trames de la requête et de la réponse d'une commande Modbus sont décrits, les uns après les autres, à l'exception des champs représentés ci-contre. Ceux-ci sont systématiquement présents dans les requêtes et les réponses de toutes les commandes Modbus.

Les champs « Slave Address » et « Fonction » constituent les deux premiers octets de ces trames. Les deux octets du « Checksum » constituent leurs deux derniers octets.

Slave Address	- Valeur non modifiable (adresse Modbus : 1 à 247 ; adresses 65, 126 et 127 réservées)
Fonction	- Valeur non modifiable (code de la commande Modbus)
... Autres champs ...	... Spécificités des commandes Modbus ...
Cheksum (Lo)	- Type du contrôle d'erreur - N° du 1er octet contrôlé
Cheksum (Hi)	

Les descriptions des trames Modbus qui figurent dans les chapitres suivants sont principalement destinées à vous aider à configurer les échanges Modbus de la passerelle à l'aide de ABC-LUFP Config Tool. Reportez-vous à la documentation des esclaves Modbus pour prendre connaissance des limites d'utilisation de ces trames pour chacun d'eux (nombre de registres pouvant être lus ou écrits en une seule commande Modbus, par exemple).

Il est préférable que vous vous procuriez un document Modbus standard, tel que le guide intitulé *Modicon Modbus Protocol Reference Guide* (réf. : PI-MBUS-300 Rev. J), afin de pouvoir faire la correspondance entre les éléments affichés dans ABC-LUFP Config Tool et le contenu des trames Modbus correspondantes. Voici un exemple de correspondance pour une trame complète (y compris les champs de début et de fin de trame présentés ci-dessus), basée sur la commande Read Holding Registers.

	Éléments sous ABC-LUFP Config Tool	Champs des trames Modbus	Taille
<b>Requête Modbus</b>	Slave Address	Slave Address	1 octet
	Function Code	Function Code	1 octet
	Starting register address	Starting Address	2 octets
	Number of registers	Quantity of Registers	2 octets
	Checksum	CRC16	2 octets
<b>Réponse Modbus</b>	Slave Address	Slave Address	1 octet
	Function Code	Function Code	1 octet
	Byte count	Byte Count	1 octet
	Data	First Register Value	2 octets
		.....	.....
		Last Register Value	2 octets
Checksum	CRC16	2 octets	

## Annexe E : Commandes Modbus

Le chapitre 6.12 présente lui aussi quelques exemples de correspondance entre les éléments affichés dans ABC-LUFP Config Tool et les champs des trames Modbus correspondantes.

**Voir également :** Chapitre 6.12.2, et chapitre 6.12.3, dans le cas où l'implémentation de l'une de ces commandes serait incompatible avec son implémentation dans la passerelle, par exemple. Il devient alors nécessaire de créer une commande Modbus spéciale afin de palier cette incompatibilité.

### Commande « Read Holding Registers » (0x03)

Trame	Champ ABC-LUFP Config Tool	Valeur ou propriétés
Requête	Starting Register Address	- Adresse du registre
	Number of Registers	- Nombre de registres
	Checksum	- CRC16
Réponse	Byte Count	- Nombre d'octets de données = nombre de registres × 2
	Data (premier registre)	- Byte swap = « Swap 2 bytes »
	.....	- Data length = Valeur du champ « Byte count »
	Data (dernier registre)	- Data location = Adresse dans la mémoire d'entrée de la passerelle
	Checksum	- CRC16

### Commande « Preset Single Register » (0x06)

Trame	Champ ABC-LUFP Config Tool	Valeur ou propriétés
Requête	Register Address	- Adresse du registre
	Preset Data	- Byte swap = « Swap 2 bytes » - Data length = 0x0002 - Data location = Adresse dans la mémoire de sortie de la passerelle
Réponse	Register Address	- Byte swap = « Swap 2 bytes » - Data length = 0x0002
	Preset Data	- Data location = Adresse dans la mémoire d'entrée de la passerelle
	Checksum	- CRC16

**REMARQUE :** Etant donné que la réponse de l'esclave fait écho à la requête, vous n'avez pas besoin de mapper le champ « Preset Data » de la réponse dans la zone mémoire d'entrée (0x0200-0x03FF) si vous considérez que cet écho est inutile pour le maître DeviceNet. Au lieu de cela, vous pouvez le mapper dans la zone mémoire générale (en commençant à l'adresse 0x0400).

# Annexe E : Commandes Modbus

## Commande « Preset Multiple Registers » (0x10)

Trame	Champ ABC-LUFP Config Tool	Valeur ou propriétés
Requête	Starting Register Address	- Adresse du 1er registre
	Number of Registers	- Nombre de registres
	Byte Count	- Nombre d'octets de données = nombre de registres × 2
	Data (premier registre)	- Byte swap = « Swap 2 bytes »
	.....	- Data length = Valeur du champ « Byte count »
	Data (dernier registre)	- Data location = Adresse dans la mémoire de sortie de la passerelle
	Checksum	- CRC16
Réponse	Starting Register Address	- Adresse du 1er registre
	Number of Registers	- Nombre de registres
	Checksum	- CRC16

## Réponses d'exception du protocole Modbus

Lorsqu'il est dans l'impossibilité d'exécuter une commande dictée par une requête Modbus, un esclave envoie une réponse d'exception à la place de la réponse normale à la requête.

### AVERTISSEMENT

#### FONCTIONNEMENT SANS SURVEILLANCE DU SYSTEME

Dans le cas des commandes Modbus standard, la passerelle LUFP9 considère que toutes les réponses d'exception qu'elle reçoit de la part des esclaves Modbus sont des réponses erronées. Par conséquent, elle effectuera les ré-émissions configurées pour les requêtes incriminées.

Si vous souhaitez que le logiciel applicatif de votre maître DeviceNet puisse gérer les réponses d'exception d'une manière spécifique, vous avez la possibilité de remplacer la commande Modbus, dans ABC-LUFP Config Tool, par une commande personnalisée (voir chapitre 6.12.3.2). Cela permet alors de remonter les champs « Slave Address » et « Function » jusqu'au maître DeviceNet.

**Le non-respect de ces instructions peut entraîner la mort, de graves blessures ou des dommages matériels.**

La structure d'une réponse d'exception est indépendante de la commande Modbus associée au champ « Function » de la requête incriminée. L'intégralité de la trame d'une réponse d'exception est présentée ci-dessous :

Slave Address	Adresse Modbus (1 à 247 ; adresses 65, 126 et 127 réservées) : La valeur de ce champ est identique à celle du champ « Slave Address » de la requête incriminée.
Function	Code de la commande, avec indicateur d'exception : La valeur de ce champ est égale à 0x80 + la valeur du champ « Function » de la requête incriminée.
Exception Code	Code indiquant la nature de l'erreur qui est à l'origine de la réponse d'exception (voir tableau présenté sur la page suivante).
Cheksum (Lo)	Contrôle d'erreur.
Cheksum (Hi)	

## Annexe E : Commandes Modbus

---

Code	Nom de l'exception	Description de l'exception
0x01	ILLEGAL FUNCTION	La commande « Fonction » de la requête n'est pas implémentée dans le logiciel de l'esclave Modbus, ou bien celui-ci n'est pas en mesure de l'exécuter pour l'instant.
0x02	ILLEGAL DATA ADDRESS	La combinaison des champs « Starting Address » et « No. of Registers » de la requête (ou champs assimilés) donne accès à une ou plusieurs adresses non accessibles sur l'esclave Modbus.
0x03	ILLEGAL DATA VALUE	La valeur de l'un des champs de la requête Modbus est hors limites autorisées. Cette erreur ne concerne pas le contenu des champs « Data » (ou assimilés), car elle ne tient compte que des champs utiles à la gestion du protocole Modbus.
0x04	SLAVE DEVICE FAILURE	Une erreur irrémédiable s'est produite lors de l'exécution de la commande.
0x05 (1)	ACKNOWLEDGE	L'esclave Modbus informe la passerelle qu'il a pris en compte la commande (acquiescement), mais que son exécution est trop longue pour qu'il puisse se permettre d'attendre qu'elle soit menée à terme avant de pouvoir émettre une réponse. La passerelle devra émettre des requêtes ultérieures afin de déterminer si la commande est achevée ou non.
0x06 (1)	SLAVE DEVICE BUSY	L'esclave Modbus informe la passerelle qu'il est déjà en train d'exécuter une commande et qu'il ne peut donc pas exécuter celle qui lui est transmise. La passerelle devra donc ré-émettre la requête ultérieurement.
0x07 (1)	NEGATIVE ACKNOWLEDGE	L'esclave Modbus informe la passerelle qu'il n'est pas en mesure d'exécuter la commande demandée. Cette exception ne concerne que les commandes 13 et 14 (0x0D et 0x0E). Ces fonctions ne font pas partie des commandes Modbus standard et ne sont pas décrites dans ce document.
0x08 (1)	MEMORY PARITY ERROR	L'esclave Modbus informe la passerelle qu'il a détecté une erreur de parité lors de l'accès à sa propre mémoire. Cette exception ne concerne que les commandes standard 20 et 21 (0x14 et 0x15). Ces commandes ne sont pas supportées par la passerelle.

- (1) Reportez-vous à la documentation Modbus standard pour de plus amples renseignements au sujet de ces différents cas de figure.

# Index

---

## A

Adresse, 22  
Adresse MAC ID, 32  
Allen Bradley  
  SLC500, 37  
Architecture, 9, 25  
Automate maître DeviceNet, 31

## B

Boîte de dérivation SCA TSXCA50, 19  
Boîte de dérivation VW3 A8 306 TF3, 19  
Boîtes de dérivation SCA, 17

## C

Câble Modbus, 19  
Câble VW3 A68 306, 17  
Communications  
  apériodiques, 35, 36, 37  
  périodiques, 35, 36, 37  
Commutateur, 21  
Connecteur RJ45, 17

## D

DEL, 23  
Diagnostic LEDs, 12  
Documents associés, 5  
Données échangées, 11  
Double terminaison VW3 A8 306 RC, 19

## E

Esclave DeviceNet, 10  
Esclaves Modbus, 9, 10

## F

Fichier EDS, 31

## P

Paramètres, 32  
Prise abonnés 2 voies TSXSCA62, 19  
Protective Earth, 13

## R

Rail DIN, 13  
Répartiteur LU9GC03, 19  
Résistance de ligne, 20  
RSLogix 500, 37  
RSNetWorx, 35, 36

## S

Scanner DeviceNet, 34  
SLC500, 37

## T

Temps de cycle, 26  
Topologie  
  bus, 15, 16

## V

Vitesse de communication, 21



# Glossaire

---

<b>0x****</b>	Valeur exprimée au format hexadécimal, ce qui équivaut aux notations H****, ****h et 16#**** parfois utilisées dans d'autres documents. <b>REMARQUE</b> : Le logiciel ABC-LUFP Config Tool utilise la notation 0x**** notation. Exemple : 0x0100 = 16#0100 = 256.
<b>2#**** ****</b>	Valeur exprimée en binaire. Le nombre de digits '•' dépend de la taille de la donnée représentée. Chaque quartet (groupe de 4 bits) est séparé des autres quartets par un espace. Exemple octet 2#0010 0111 = 39, mot 2#0110 1001 1101 0001 = 0x69D1 = 27089.
<b>ABC-LUFP Config Tool</b>	Nom du logiciel PC utilisé pour configurer et surveiller la passerelle DeviceNet/Modbus LUFP9.
<b>ATS</b>	Abréviation de « Altistart » (démarrreur).
<b>ATV</b>	Abréviation de « Altivar » (variateur de vitesse).
<b>CRC</b>	Cyclical Redundancy Check.
<b>DEL</b>	Diode Electro-Luminescente.
<b>EDS</b>	Electronic Data Sheet. Désigne le format des fichiers (extension « .eds ») qui permettent à un outil de configuration et de mise au point de maîtres DeviceNet de configurer leurs échanges selon ce même protocole.
<b>Fieldbus</b>	Terme désignant le réseau amont DeviceNet dans ABC-LUFP Config Tool.
<b>Handshake</b>	Ancien terme désignant les deux registres d'initialisation et de diagnostic de la passerelle LUFP9. Ce terme a été remplacé par l'expression « Control/Status Byte ».
<b>LRC</b>	Longitudinal Redundancy Check.
<b>LSB</b>	Octet de poids faible d'un mot de 16 bits.
<b>MAC ID</b>	Media Access Control ID. Adresse d'un module sur un bus DeviceNet.
<b>MSB</b>	Octet de poids fort d'un mot de 16 bits.
<b>Nœud</b>	Terme désignant le point de connexion d'un esclave Modbus dans ABC-LUFP Config Tool.
<b>ODVA</b>	Open DeviceNet Vendor Association, Inc.
<b>PSU</b>	Alimentation
<b>Sub-Network</b>	Terme désignant le réseau aval DeviceNet dans ABC-LUFP Config Tool.
<b>XML</b>	EXtensible Markup Language. Langage utilisé par ABC-LUFP Config Tool pour l'importation/exportation de la configuration d'un esclave Modbus.