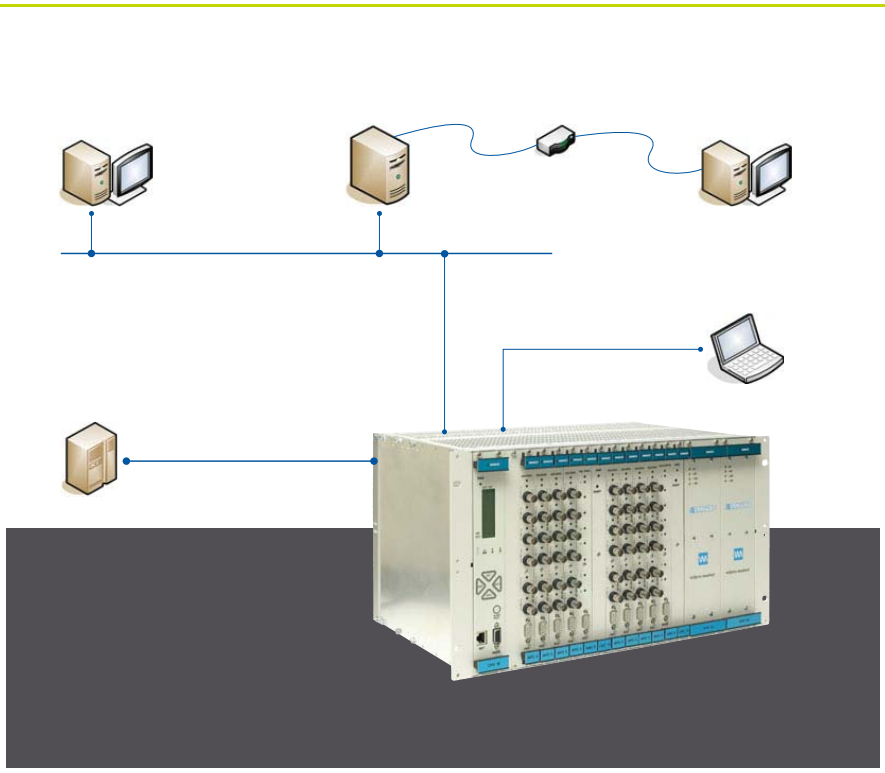
Enabling the Extraordinary
To Fly To Power To Live

MEGGiTT

MANUEL

vibro-meter®

# Mise en réseau du VM600



Référence de document MAVM600-NET/F
Édition 10 – Février 2021

Meggitt SA, Route de Moncor 4, Case postale, 1701 Fribourg, Switzerland

Tel: +41 26 407 1111
Fax: +41 26 407 1660
energy@ch.meggitt.com
www.meggittsensing.com/energy
www.meggitt.com

# FEUILLE DE SUIVI DES RÉVISIONS

| Édition | Date de publication | Rédigé par / modifié par | Description | Signature |
|---|---|---|---|---|
| 1 | 07.10.03 | R. Meyer | Édition originale | RM |
| 2 | 31.03.06 | N. Parker | Mise à jour destinée à refléter la nouvelle identité d'entreprise | NP |
| 3 | 13.05.09 | S. Trono | Mis à jour pour Windows XP<br>Ajout du support de la carte AMC8 | ST |
| 4 | 02.09.2010 | Peter Ward | Mise à jour destinée à refléter le nouveau support Modbus disponible pour les cartes CPUM exécutant la version logicielle 071 ou ultérieure. Ceci concerne tout particulièrement les cartes AMC8 et MPC4 et leurs tableaux de définition du registre, qui ont été placées en annexe.<br>Mise à jour également destinée à refléter le nouveau support PROFINET disponible pour les cartes CPUM exécutant la version logicielle 801 ou ultérieure. | PW |
| 5 | 24.11.2010 | Peter Ward | Rectifications mineures | PW |
| 6 | 15.03.2012 | Peter Ward | Rectifications à le Table A-2 sur la page A-4.<br>Mis à jour selon l'identité de marque Meggitt. | PW |
| 7 | 30.06.2014 | Peter Ward | Ajout d'informations complémentaires sur le protocole DHCP (dynamic host configuration protocol).<br>Correction du débit en bauds Modbus par défaut dans la section 8.4.2 Paramètres de communication du VM600.<br>Mise à jour à partir des dernières informations de mise en réseau, y compris l'utilisation du logiciel CPUM Configurator pris en charge par la version 075 du microprogramme CPUM et des sessions de terminal (avec PuTTY et Midnight Commander). Cette mise à jour a nécessité la restructuration des sept premiers chapitres du manuel.<br>Note : Pas disponible en français. | PW |
| 8 | 08.05.2015 | Peter Ward | Mise à jour des tableaux dans la Préface pour inclure des informations complémentaires sur les cartes CPUM (voir la section Produits concernées).<br>Mise à jour pour tenir compte du fait que le logiciel CPUM Configurator est désormais inclus avec la version 2.7 ou ultérieure du logiciel VM600 MPSx (voir la section 6 CPUM Configurator).<br>Clarification et correction du texte d'introduction à PuTTY (voir la section 1.3.2.1 PuTTY).<br>Mise à jour des références croisées incorrectes et correction des fautes de frappe dans l'ensemble du texte.<br>Note : Pas disponible en français. | PW |

| Édition | Date de publication | Rédigé par / modifié par | Description | Signature |
|---|---|---|---|---|
| 9 | 14.02.2018 | Peter Ward | Ajout d'une déclaration stipulant que les certifications et garanties de produits Meggitt SA sont valables uniquement pour les produits achetés directement auprès de Meggitt SA ou d'un distributeur agréé (voir la section AVIS IMPORTANTS).<br>Mise à jour des tableaux dans la Préface pour inclure des informations complémentaires sur les cartes CPUM (voir la section Produits concernées).<br>Mise à jour pour tenir compte des modifications apportées au logiciel CPUM Configuration inclus avec VM600 MPSx 2.7 build 012 ou version ultérieure (voir 6 CPUM Configurator).<br>Mise à jour pour inclure la version 081 du microprogramme CPUM qui ajoute la prise en charge de PROFINET (voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM et au chapitre 10 Configuration d'une connexion PROFINET (logiciel CPUM version 081 ou ultérieure)).<br>Ajout d'informations sur l'élimination des produits en fin de vie (voir la section 11 Élimination des produits en fin de vie).<br>Mise à jour de la procédure et du formulaire de retour d'un produit énergétique conformément aux informations du site Web de Meggitt vibro-meter® (voir la section 12 Assistance client). | PW |
| 10 | 03.02.2021 | Peter Ward | Ajout d'une note précisant que la prise en charge de la carte CPUM pour PROFIBUS utilise une carte supplémentaire VM600 « Molex/VME » pour fournir l'interface physique PROFIBUS DP (voir Préface).<br>Mise à jour de la Figure 10-2, la Figure 10-17, la Figure 10-18, la Figure 10-19, la Figure 10-20 et la Figure 10-21.<br>Mise à jour pour utiliser la dernière identité de marque Meggitt. | PW |

| | Département | Nom | Date | Signature |
|---|---|---|---|---|
| Contenu technique approuvé par | Développement | Igor Karpekin | 14.02.2018 | IK |
| | Gestion de produits | Michaël Hafner | 03.02.2021 | MH |
| Document émis par | Publications techniques | Peter Ward | 03.02.2021 | PW |

*La copie-maître dûment signée de cette page est conservée par le département des publications techniques de Meggitt SA et peut être délivrée sur demande écrite adressée au directeur des publications techniques.*

## AVIS IMPORTANTS

Toutes les déclarations, informations techniques et recommandations de ce document relatives aux produits fournis par Meggitt vibro-meter® (Meggitt SA) s'appuient sur des informations jugées fiables, Nous ne saurions toutefois en garantir l'exactitude ou le caractère exhaustif, sauf stipulation contraire expressément notifiée par Meggitt SA. Avant d'utiliser ce produit, vous devez l'évaluer et déterminer s'il convient à l'application envisagée. Vous devez également consulter notre site web à l'adresse www.meggittsensing.com/energy pour toute mise à jour des fiches techniques, des certificats Ex, des schémas de produits, des manuels d'utilisation, des bulletins de service et/ou d'autres instructions affectant le produit.
Sauf convention expresse écrite contraire avec Meggitt SA, vous assumez tous les risques et responsabilités associés à l'utilisation du produit. Meggitt SA décline toute responsabilité pour toute déclaration relative au produit qui ne figurerait pas dans une publication en anglais actuelle de Meggitt SA (Meggitt) ou pour toute déclaration présente dans des extraits, des résumés, des traductions ou tout autre document non écrit/produit par Meggitt SA. Les certifications et garanties applicables aux produits fournis par Meggitt SA sont valables uniquement pour les nouveaux produits achetés directement auprès de Meggitt SA ou d'un distributeur autorisé de Meggitt SA. Meggitt SA se réserve le droit de modifier toute partie de cette publication sans préavis.

## CONTRÔLE DES EXPORTATIONS

Les informations contenues dans ce document peuvent être soumises aux réglementations de contrôle des exportations de la Communauté européenne, des États-Unis ou d'autres pays. Chaque destinataire de ce document est tenu de s'assurer que le transfert ou l'utilisation de toute information contenue dans ce document est conforme à toutes les réglementations de contrôle des exportations pertinentes. ECN N/A.

## COPYRIGHT

Copyright© 2009-2021 Meggitt SA

Tous droits réservés.

Publié et imprimé par Meggitt SA à Fribourg, Suisse

Les noms des sociétés et des produits mentionnés
dans ce document peuvent être les marques de commerce de leurs propriétaires respectifs.

Les informations contenues dans ce document sont susceptibles d'être modifiées sans préavis.
Ces informations ne doivent pas être utilisées, reproduites ou publiées que ce soit en totalité ou en partie, sans l'autorisation écrite expresse de Meggitt SA (Meggitt vibro-meter®).

# PRÉFACE

## À propos de ce manuel

Ce manuel fournit des informations de référence sur l'utilisation de systèmes de surveillance à base de racks VM600, de la gamme de produits Meggitt vibro-meter®, dans des réseaux. Il s'applique aux systèmes de protection et aux systèmes de surveillance à base de racks VM600 suivants :

- Système de protection des machines (MPS)
- Système de contrôle de l'état (CMS).

## À propos de Meggitt et vibro-meter®

Meggitt PLC est un groupe d'ingénierie mondial, basée au Royaume-Uni, spécialisé dans la conception et la fabrication de composants et de systèmes haute performance pour les marchés de l'aérospatiale et de l'énergie.

Le site Meggitt de Fribourg, en Suisse, exerce ses activités en tant qu'entité légale Meggitt SA (anciennement Vibro-Meter SA). vibro-meter® est une ligne de produits de Meggitt qui applique nos technologies de détection et de surveillance de base à la production d'énergie, au pétrole et au gaz et à d'autres marchés industriels.

Meggitt SA propose une large gamme de capteurs de vibrations, de pression dynamique, de proximité, d'entrefer et autres capables de fonctionner dans des environnements extrêmes, de systèmes de surveillance et de protection électroniques, ainsi que de logiciels innovants pour les turbomachines aérospatiales et terrestres.

Les produits et solutions vibro-meter® sont à la pointe de la détection et de la surveillance depuis plus de 65 ans et contribuent à ce que les machines et les équipements continuent de fonctionner de manière sûre, fiable et efficace. Ceci inclut les systèmes de surveillance à base de racks VM600 produits pour la gamme de produits vibro-meter®.

Pour en savoir plus sur Meggitt Suisse, notre fière tradition d'innovation et d'excellence et nos solutions pour les marchés de l'énergie et leurs applications, visitez le site web à l'adresse www.meggittsensing.com/energy

## À qui s'adresse ce manuel ?

Ce manuel a été rédigé pour les opérateurs de systèmes de commande/contrôle de processus utilisant des systèmes à base de racks VM600.

L'opérateur est supposé avoir la formation technique nécessaire en électronique et en mécanique (certificat professionnel/diplôme ou équivalent) pour lui permettre d'installer, de programmer et d'utiliser le système.

## Produits concernées

Les tableaux suivants donnent un aperçu des cartes CPUM concernées par ce manuel, en soulignant certaines de leurs caractéristiques et fonctionnalités principales.

| Numéro de commande CPUM (PNR)) | Mémoire | | Système d'exploitation | Fonctionnalités | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Mémoire On Chip (Mo) | Compact Flash (Mo) | Version QNX | Protocole PPP activé | Modbus | Valeur discrète codée dans le registre analogique | PROFIBUS* | PROFINET | Sécurité (CPUM) du rack VM600 MPS |
| 200-595-04x-**C**1x | 16 | | 4.25 | | | | | | |
| 200-595-061-**C**14<br>200-595-062-**C**14 | 16 | | 6.0.3 | | | | | | |
| 200-595-063-**C**14<br>200-595-064-**C**14<br>200-595-065-**C**14 | 32 | | 6.0.3 | ✓<br>✓ | | <br>✓ | | | |
| 200-595-066-**C**14 | | 512 | 6.0.3 | ✓ | | ✓ | | | |
| 200-595-067-**C**14 | | 512 | 6.0.3 | ✓ | ✓ | ✓ | | | |
| 200-595-067-**C**2x<br>200-595-068-**C**2x<br>200-595-070-**C**2x<br>200-595-071-**C**2x<br>200-595-072-**C**2x<br>200-595-073-**C**2x | | 512 | 6.0.3 | ✓<br>✓<br>✓<br>✓<br>✓ | ✓<br>✓<br>✓<br>✓<br>✓<br>✓ | ✓<br>✓<br>✓<br>✓<br>✓<br>✓ | <br><br><br><br><br>✓ | | |
| 200-595-801-**C**2x | | 512 | 6.4.1 | ✓ | ✓ | ✓ | | ✓ | |
| 200-595-074-**C**2x<br>200-595-075-**C**2x | | 512 | 6.4.1 | | ✓<br>✓ | ✓<br>✓ | ✓<br>✓ | | |
| 200-595-076-**D**xx<br>200-595-077-**D**xx<br>200-595-078-**D**xx<br>200-595-079-**D**xx | | 512 | 6.4.1 | | ✓<br>✓<br>✓<br>✓ | ✓<br>✓<br>✓<br>✓ | ✓<br>✓<br>✓<br>✓ | | <br>✓<br>✓<br>✓ |
| 200-595-080-**D**xx | | 512 | 6.4.1 | | ✓ | ✓ | ✓ | | ✓ |
| 200-595-081-**D**xx | | 512 | 6.4.1 | | ✓ | ✓ | ✓ | ✓ | ✓ |

Note
Voir les notes du tableau à la page vii.



MEGGiTT

| Numéro de commande CPUM (PNR) | Mémoire | | Système d'exploitation | Fonctionnalités | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Mémoire On Chip (Mo) | Compact Flash (Mo) | Version QNX | Protocole PPP activé | Modbus | Valeur discrète codée dans le registre analogique | PROFIBUS* | PROFINET | Sécurité (CPUM) du rack VM600 MPS |
| 200-595-61x-**C**1x<br>200-595-62x-**C**1x | 16 | | 4.25 | | | | | | |
| 200-595-701-**C**14 | 32 | | 6.0.3 | ✓ | | | | | |
| 200-595-703-**C**2x | | 512 | 6.0.3 | ✓ | ✓ | ✓ | | | |
| 200-595-724-**C**2x | | 512 | 6.0.3 | | ✓ | ✓ | ✓ | | |
| 200-595-72x-**C**2x<br>200-595-750-**C**2x | | 512 | 6.4.1 | | ✓<br>✓ | ✓<br>✓ | ✓<br>✓ | | |

Notes

Les versions standard disponibles sur le marché de la carte CPUM exécutent généralement des versions « standard » du microprogramme et sont généralement préconfigurées avec une configuration « standard ».

Les versions spécifiques au client de la carte CPUM exécutent généralement des versions « spéciales » du microprogramme qui assurent la prise en charge de fonctions spécifiques au client et/ou qui sont préconfigurées avec une configuration spécifique au client. Contactez Meggitt SA pour plus d'informations.

Dans la colonne **Numéro de commande CPUM (PNR)**, la lettre **C** indique les versions suivantes de la carte CPUM :

**1** Carte CPU modulaire avec une interface Ethernet et une interface série (ports de communication Ethernet et série intégrés fournissant une connexion RS-232 ou RS-485/422 et un contrôleur Ethernet intégré).

**3** Carte CPU modulaire avec deux interfaces Ethernet et une interface série (ports de communication Ethernet et série intégrés, avec un module contrôleur Ethernet supplémentaire fournissant une connexion Ethernet redondante).

**4** Carte CPU modulaire avec une interfaces Ethernet et deux interfaces série (ports de communication Ethernet et série intégrés, avec un module de communication série supplémentaire fournissant une connexion RS-485/422 redondante.

La modification du numéro de commande CPUM (PNR) de 200-595-075-C2x à 200-595-076-Dxx correspond à la mise à jour de la carte CPUM en 2015 pour utiliser un nouveau module CPU PC/104 (PFM-541I ou équivalent) qui prend en charge deux interfaces Ethernet par défaut, ce qui nécessite des mises à jour de la carte support sous-jacente. (Auparavant, la carte CPUM était équipée d'un module CPU (MSM586SEN ou équivalent) prenant en charge une interface Ethernet par défaut et nécessitait l'installation d'un module de contrôleur Ethernet supplémentaire (MSME104 ou équivalent) pour assurer une connexion Ethernet redondante.)

Dans la colonne **Numéro de commande CPUM (PNR)**, la lettre **D** indique les versions suivantes de la carte CPUM :

**3** Versions Ethernet redondantes de la carte CPUM, c'est-à-dire avec deux interfaces Ethernet (et deux interfaces série).

**5** Versions redondantes en série de la carte CPUM, c'est-à-dire avec deux interfaces Ethernet (et deux interfaces série) et deux interfaces série supplémentaires.

Pour une explication de la colonne **Valeur discrète codée dans le registre analogique**, voir la section 8.6.5 Valeurs discrètes codées dans un registre de valeurs analogiques.

*La colonne **PROFIBUS** indique la prise en charge de la carte CPUM pour PROFIBUS en utilisant une carte supplémentaire VM600 « Molex/VME » pour fournir l'interface PROFIBUS DP physique.

## Structure du manuel

Cette section présente un aperçu de la structure du document et des informations qu'il contient. Certaines informations ont été volontairement répétées dans différentes sections du document pour réduire les références croisées et faciliter la compréhension.

Les chapitres sont présentés dans un ordre logique. Vous devez lire ceux qui vous concernent plus particulièrement et garder le document à portée de main pour pouvoir vous y référer ultérieurement.

Le document se présente comme suit :

**Chapter 1** Introduction et présentation de la mise en réseau du VM600

Fournit une introduction au travail avec des racks VM600 en réseau, ainsi que des options de communication et des recommandations. Des informations générales sur les réseaux (Ethernet), les termes et les principes informatiques sont également fournies.

**Chapter 2** Fichiers de configuration et structure des répertoires de la carte CPUM

Fournit une introduction aux différents fichiers qui définissent la configuration et le fonctionnement de la carte CPUM et du rack VM600, et explique comment travailler avec ces fichiers.

**Chapter 3** Communications série avec un rack VM600

Décrit comment établir des communications série avec un rack VM600 (carte CPUM) et effectuer des tâches de base. Par exemple :

- Paramètres de communication série par défaut.
- Comment identifier la version du microprogramme d'une carte CPUM.
- Comment découvrir l'adresse IP d'une carte CPUM.

**Chapter 4** Configuration d'une connexion Ethernet

Fournit des informations sur la configuration d'un ordinateur pour communiquer avec un rack VM600 (carte CPUM) sur une liaison Ethernet. Par exemple :

- Paramètres de communication Ethernet par défaut (adresse IP).
- Comment configurer un ordinateur pour les communications avec une carte CPUM.
- Comment tester les communications.
- Communications Ethernet sur la carte CPUM principale et secondaire.

**Chapter 5** Communications Ethernet avec un rack VM600

Décrit comment établir des communications Ethernet (TCP/IP) avec un rack VM600 (carte CPUM) et effectuer des tâches de base. Par exemple :

- Comment établir des communications Ethernet avec une carte CPUM.
- Comment vérifier l'adresse IP d'une carte CPUM.
- Comment modifier l'adresse IP d'une carte CPUM et modifier d'autres fichiers de configuration.
- Comment ajouter une passerelle par défaut à une carte CPUM.

**Chapter 6** CPUM Configurator

Décrit comment utiliser le logiciel CPUM Configurator pour configurer et gérer un rack VM600 sur une liaison Ethernet. Par exemple :

- Comment modifier l'adresse IP d'une carte CPUM.
- Comment transférer des fichiers de configuration et travailler avec.

**Chapter 7** Émulation de terminal

Décrit comment utiliser une session d'émulation de terminal, via PuTTY et Midnight Commander, pour configurer et gérer un rack VM600 sur une liaison Ethernet ou série. Par exemple :

- Comment modifier l'adresse IP d'une carte CPUM.
- Comment transférer des fichiers de configuration et travailler avec.

**Chapter 8** Configuration d'une connexion Modbus (logiciel CPUM version 067 ou antérieure)

Décrit les protocoles Modbus RTU et Modbus TCP. Fournit des informations détaillées sur la lecture des entrées discrètes et des registres du rack VM600.

**Chapter 9** Configuration d'une connexion Modbus (logiciel CPUM version 071 ou ultérieure)

Fournit des informations supplémentaires détaillées sur les fonctionnalités ajoutées par les améliorations apportées à l'interface logicielle Modbus.

**Chapter 10** Configuration d'une connexion PROFINET (logiciel CPUM version 081 ou ultérieure)

Fournit une description de base de PROFINET et des informations détaillées sur la configuration d'un ordinateur exécutant SIMATIC STEP7 pour communiquer avec un VM600 au moyen de PROFINET.

**Chapter 11** Élimination des produits en fin de vie

Fournit des informations et des coordonnées concernant l'élimination écologique des équipements électriques/électroniques à la fin de leur vie utile.

**Chapter 12** Assistance client

Fournit les coordonnées des services à contacter pour les questions techniques et les informations concernant la réparation et le retour de l'équipement.

**NOTE :** Si vous utilisez la version 071 ou ultérieure du microprogramme CPUM pour implémenter des communications Modbus, il est important de lire à la fois le **Chapter 8** et le **Chapter 9** car le chapitre 9 Configuration d'une connexion Modbus (logiciel CPUM version 071 ou ultérieure) s'appuie sur les informations fournies dans le **Chapter 8**.

**NOTE :** Si vous utilisez la version 081 ou ultérieure du microprogramme CPUM pour implémenter des communications PROFINET, il est important de lire également le **Chapter 8** et le **Chapter 9** car le chapitre 8 Configuration d'une connexion Modbus (logiciel CPUM version 067 ou antérieure) s'appuie sur les informations fournies dans le **Chapter 8** et dans le **Chapter 9**.

## Publications et documentation connexes

Pour plus d'informations, l'utilisateur est invité à se reporter à l'un ou plusieurs des manuels de logiciels suivants :

- *VM600 condition monitoring system (CMS) software manual – volumes 1 and 2* (référence de document MACMS-SW/E)
- *MPS1 configuration software for VM600 machinery protection systems* (référence de document MAMPS1-SW/E)
- *MPS2 configuration software for VM600 machinery protection systems* (référence de document MAMPS2-SW/E).

L'utilisateur est également invité à se reporter à l'un ou plusieurs des manuels de matériels suivants :

- *VM600 condition monitoring system (CMS) hardware manual* (référence de document MACMS-HW/E)
- *VM600 machinery protection system (MPS) – hardware manual (standard version)* (référence de document MAMPS-HW/E)
- *VM600 machinery protection system (MPS) – hardware manual (CSA version)* (référence de document° MAMPS-HW/E-CSA).

**NOTE :** Pour vous assurer d'utiliser la dernière version de la documentation, consultez le site web de Meggitt vibro-meter® à l'adresse www.meggittsensing.com/energy et vérifiez les mises à jour. Vous pouvez également contacter votre représentant Meggitt local.

## Abréviations

Le tableau suivant définit les abréviations figurant dans ce manuel, ainsi que dans la documentation Meggitt associée.

| Terme | Définition |
|---|---|
| ADC | Analog to Digital Converter (convertisseur analogique-numérique) |
| API | Automate programmable industriel |
| bps | bits per second (bits par seconde) |
| CMS | Condition Monitoring System (système de contrôle de l'état) |
| CRC | Cyclic Redundancy Check (contrôle par redondance cyclique) |
| DCS | Distributed Control System (système de commande distribué) |
| FSD | Full Scale Deflection (pleine échelle) |
| GSDML | Generic Station Description Markup Language (langage GSDML) |
| IP | Internet Protocol (protocole Internet) |
| IPX | Internetwork Packet Exchange (échange de paquets inter-réseaux) |
| LAN | Local Area Network (réseau local) |
| MPS | Machinery Protection System (système de protection des machines) |
| NetBEUI | Un protocole réseau propriétaire de Microsoft® |
| NIC | Network Interface Card (carte réseau) |
| NOS | Network Operating System (système d'exploitation du réseau) |
| PLL | Phase Locked Loop (boucle à verrouillage de phase) |
| PPP | Point-to-Point Protocol (protocole point à point) |
| RAS | Remote Access Service (service d'accès distant) |
| RTU | Remote Terminal Unit (terminal distant (Modbus)) |
| SCADA | Supervisory Control and Data Acquisition (télésurveillance et acquisition de données) |
| SIMATIC | Système et logiciel d'automatisation développé par Siemens |
| SLIP | Serial Line Internet Protocol (protocole Internet de liaison série) |
| STP | Shielded Twisted Pair (paire torsadée blindée) |
| TCP/IP | Transmission Control Protocol / Internet Protocol (protocole de contrôle de transmission / protocole Internet) |
| UTP | Unshielded Twisted Pair (paire torsadée non blindée) |
| VT100 | Un terminal vidéo fabriqué par Digital Equipment Corporation (DEC) qui est devenu la référence en matière d'émulateurs de terminaux |
| WAN | Wide Area Network (réseau étendu) |
| XML | eXtensible Markup Language (langage XML) |

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# TABLE DES MATIÈRES

**TITLE PAGE** . . . . . i

**FEUILLE DE SUIVI DES RÉVISIONS** . . . . . ii

**AVIS IMPORTANTS** . . . . . iv

**PRÉFACE** . . . . . v

**TABLE DES MATIÈRES** . . . . . xiii

**1 INTRODUCTION ET PRÉSENTATION DE LA MISE EN RÉSEAU DU VM600** . . . . . 1-1

1.1 Possibilités de communication . . . . . 1-1
1.2 Recommandations pour la communication . . . . . 1-4
1.3 Logiciels et protocoles de communication . . . . . 1-5
1.3.1 CPUM Configurator . . . . . 1-5
1.3.2 Telnet . . . . . 1-5
1.3.2.1 PuTTY . . . . . 1-6
1.3.2.2 Client FileZilla . . . . . 1-6
1.3.2.3 Notepad++ . . . . . 1-6
1.3.3 Midnight Commander . . . . . 1-7
1.3.4 Éditeur vi . . . . . 1-7
1.4 Quelques termes et définitions de la mise en réseau . . . . . 1-7
1.4.1 Types de réseau . . . . . 1-7
1.4.1.1 Réseaux locaux . . . . . 1-7
1.4.1.2 Réseaux étendus . . . . . 1-8
1.4.2 Connectivité au sein des réseaux . . . . . 1-9
1.4.3 Ethernet . . . . . 1-11
1.4.4 Protocoles réseau . . . . . 1-12
1.4.4.1 Protocole TCP/IP . . . . . 1-12
1.4.4.2 Protocole DHCP (dynamic host configuration protocol) . . . . . 1-12
1.4.5 Adressage IP . . . . . 1-13
1.4.5.1 Règles de base . . . . . 1-13
1.4.5.2 Adressage IP et classes d'adresses . . . . . 1-13
1.4.5.3 Sous-réseaux et masques de sous-réseaux . . . . . 1-15

**2 FICHIERS DE CONFIGURATION ET STRUCTURE DES RÉPERTOIRES DE LA CARTE CPUM . . . . . 2-1**

2.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme . . . . . 2-1
2.2 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme . . . . . 2-2
2.3 Fichiers de configuration Modbus . . . . . 2-2
2.3.1 Cartes CPUM exécutant la version 071 ou ultérieure du microprogramme . . . . . 2-2
2.3.2 Cartes CPUM exécutant la version 067 ou antérieure du microprogramme . . . . . 2-4
2.4 Fichier de configuration PROFINET . . . . . 2-5
2.4.1 Cartes CPUM exécutant la version 081 ou ultérieure du microprogramme . . . . . 2-5
2.4.2 Cartes CPUM exécutant la version 801 du microprogramme . . . . . 2-5
2.5 Utilisation des fichiers de configuration . . . . . 2-5
2.6 Configuration des cartes CPUM . . . . . 2-6
2.6.1 Configurations standard . . . . . 2-6
2.6.2 Configurations spécifiques au client . . . . . 2-7
2.6.3 Modification des configurations . . . . . 2-7

**3 COMMUNICATIONS SÉRIE AVEC UN RACK VM600 . . . . . 3-1**

3.1 Paramètres de communication série par défaut . . . . . 3-1
3.2 Établissement de communications série avec une carte CPUM . . . . . 3-2
3.3 Identification de la version du microprogramme d'une carte CPUM . . . . . 3-4
3.3.1 À propos du fichier version.txt . . . . . 3-5
3.3.1.1 Exemple de fichier version.txt . . . . . 3-5
3.3.1.2 Description . . . . . 3-5
3.4 Découverte de l'adresse IP d'une carte CPUM . . . . . 3-6
3.4.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme . . . . . 3-6
3.4.2 À propos du fichier net.cfg . . . . . 3-7
3.4.2.1 Exemple de fichier net.cfg . . . . . 3-7
3.4.2.2 Description du fichier net.cfg . . . . . 3-7
3.4.3 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme . . . . . 3-8

**4 CONFIGURATION D'UNE CONNEXION ETHERNET . . . . . 4-1**

4.1 Paramètres d'adresse IP par défaut . . . . . 4-1

4.1.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme . . . . . 4-2

4.1.2 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme . . . . . 4-2

4.2 Configuration d'un ordinateur pour des communications Ethernet avec un rack VM600 . . . . . 4-3

4.2.1 Configuration requise . . . . . 4-3

4.2.2 Configuration de la carte réseau d'un ordinateur . . . . . 4-4

4.2.3 Modification du fichier hosts d'un ordinateur . . . . . 4-6

4.2.4 Test des communications . . . . . 4-8

4.3 Redondance Ethernet . . . . . 4-9

4.3.1 Modification des adresses IP du rack . . . . . 4-10

4.3.1.1 Cartes CPUMexécutant la version 075 ou ultérieure du microprogramme . . . . . 4-10

4.3.1.2 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme . . . . . 4-12

4.4 Utilisation du logiciel VM600 avec un rack en réseau . . . . . 4-12

4.4.1 Logiciel VM600 MPS . . . . . 4-12

4.4.2 Logiciel VM600 CMS . . . . . 4-13

**5 COMMUNICATIONS ETHERNET AVEC UN RACK VM600 . . . . . 5-1**

5.1 Établissement de communications Ethernet avec une carte CPUM . . . . . 5-1

5.2 Utilisation d'une connexion Ethernet pour vérifier l'adresse IP d'une carte CPUM . 5-2

5.2.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme . . . . . 5-2

5.2.2 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme . . . . . 5-4

5.3 Utilisation d'une connexion Ethernet pour identifier la version du microprogramme d'une carte CPUM . . . . . 5-4

5.4 Modification des fichiers de configuration . . . . . 5-7

5.5 Passerelles . . . . . 5-7

5.6 Ajout d'une passerelle par défaut à une carte CPUM . . . . . 5-8

5.6.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme . . . . . 5-8

5.6.2 À propos du fichier sysinit . . . . . 5-8

5.6.2.1 Exemple de fichier sysinit . . . . . 5-8

5.6.2.2 Description . . . . . 5-9

5.6.3 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme . . . . . 5-9

**6 CPUM CONFIGURATOR . . . . . 6-1**

6.1 À propos de CPUM Configurator . . . . . 6-1
6.1.1 Fichiers de configuration de la carte CPUM . . . . . 6-1
6.2 Installation de CPUM Configurator . . . . . 6-3
6.3 Démarrage de CPUM Configurator . . . . . 6-3
6.4 Arrêt de CPUM Configurator . . . . . 6-4
6.5 Aide sur l'utilisation de CPUM Configurator . . . . . 6-4
6.6 Messages affichés lors de l'utilisation de CPUM Configurator . . . . . 6-4
6.6.1 Zone de texte Action status . . . . . 6-5
6.6.2 Bouton Clear Log . . . . . 6-5
6.6.3 Bouton Save Log . . . . . 6-5
6.7 Établissement de communications avec une carte CPUM . . . . . 6-5
6.8 Modification de l'adresse IP d'un rack VM600 (carte CPUM) . . . . . 6-6
6.9 Téléchargement de tous les fichiers de configuration à partir d'une carte CPUM . . 6-7
6.10 Chargement de tous les fichiers de configuration sur d'une carte CPUM . . . . . 6-7
6.11 Chargement de fichiers de configuration individuels sur une carte CPUM . . . . . 6-8
6.11.1 Chargement d'un fichier de configuration NTP sur une carte CPUM . . . . . 6-8
6.11.2 Chargement d'un fichier de configuration Modbus sur une carte CPUM . . 6-9
6.11.3 Chargement d'un fichier de configuration XMx16 sur une carte CPUM . . . 6-9
6.12 Modification de la configuration d'une carte CPUM . . . . . 6-10
6.13 Modification des options de CPUM Configurator . . . . . 6-11
6.14 Redémarrage d'une carte CPUM . . . . . 6-11
6.15 Mode ATP . . . . . 6-11

**7 ÉMULATION DE TERMINAL . . . . . 7-1**

7.1 À propos de PuTTY . . . . . 7-1
7.2 À propos de Midnight Commander . . . . . 7-1
7.3 Démarrage d'une session d'émulation de terminal à l'aide de PuTTY et établissement de communications avec une carte CPUM . . . . . 7-2
7.4 Arrêt d'une session d'émulation de terminal à l'aide de PuTTY . . . . . 7-3
7.5 Aide sur l'utilisation de PuTTY . . . . . 7-4
7.6 Démarrage de Midnight Commander . . . . . 7-4
7.7 Interface utilisateur de Midnight Commander . . . . . 7-4
7.7.0.1 Navigation dans Midnight Commander . . . . . 7-5
7.7.0.2 Raccourcis clavier de Midnight Commander . . . . . 7-5
7.7.0.3 Menus de Midnight Commander . . . . . 7-5
7.8 Fermeture de Midnight Commander . . . . . 7-6
7.9 Aide sur l'utilisation de Midnight Commander . . . . . 7-6

7.10 Modification de l'adresse IP d'un rack VM600 (carte CPUM) . . . . . 7-6
7.11 Affichage du contenu d'un fichier . . . . . 7-8
7.12 Communications série avec un rack VM600. . . . . 7-9

**8 CONFIGURATION D'UNE CONNEXION MODBUS (LOGICIEL CPUM VERSION 067 OU ANTÉRIEURE) . . . . . 8-1**

8.1 Introduction . . . . . 8-1
8.2 Modbus . . . . . 8-1
8.3 Communication . . . . . 8-2
8.3.1 Modbus RTU . . . . . 8-2
8.3.2 Modbus TCP. . . . . 8-3
8.4 Description du protocole Modbus RTU . . . . . 8-3
8.4.1 Trame et horloge . . . . . 8-3
8.4.2 Paramètres de communication du VM600 . . . . . 8-4
8.4.3 Fonctions Modbus prises en charge par le VM600 . . . . . 8-5
8.4.4 Formats de fonction Modbus . . . . . 8-6
8.4.4.1 Fonctions 01 et 02 . . . . . 8-6
8.4.4.2 Fonctions 03 et 04 . . . . . 8-8
8.4.5 Traitement des erreurs . . . . . 8-9
8.4.5.1 Erreurs de communication . . . . . 8-9
8.4.5.2 Erreurs système. . . . . 8-10
8.5 Description du protocole Modbus TCP . . . . . 8-10
8.6 Carte MPC4 . . . . . 8-11
8.6.1 Définition des registres . . . . . 8-12
8.6.2 Définition de sorties . . . . . 8-13
8.6.3 Valeurs analogiques (registres) . . . . . 8-14
8.6.4 Valeurs discrètes (bits internes, entrées discrètes) . . . . . 8-15
8.6.5 Valeurs discrètes codées dans un registre de valeurs analogiques. . . . . 8-16
8.6.6 Mise en correspondance des adresses . . . . . 8-16
8.7 Carte AMC8 . . . . . 8-16
8.7.1 Valeurs discrètes (bits internes) . . . . . 8-16
8.7.2 Valeurs discrètes (entrées discrètes) . . . . . 8-16
8.7.3 Registres de maintien. . . . . 8-17
8.7.4 Registres d'entrées. . . . . 8-19
8.7.5 Mise en correspondance des adresses . . . . . 8-19

**9 CONFIGURATION D'UNE CONNEXION MODBUS (LOGICIEL CPUM VERSION 071 OU ULTÉRIEURE) . . . . . 9-1**

9.1 Introduction . . . . . 9-1
9.1.1 Fonctions Modbus prises en charge par le VM600 – version 071 ou ultérieure . . . . . 9-1
9.2 Définition des registres . . . . . 9-2
9.2.1 La section RTUx . . . . . 9-3
9.2.2 La section TCP . . . . . 9-3
9.2.3 La section GLOBAL . . . . . 9-3
9.2.4 La section MAPPING . . . . . 9-4
9.3 Traitement des erreurs . . . . . 9-6
9.3.1 Erreurs de communication . . . . . 9-7
9.4 Carte MPC4 . . . . . 9-7
9.4.1 Valeurs analogiques (registres) . . . . . 9-7
9.4.1.1 Lecture des valeurs minimum et maximum . . . . . 9-7
9.4.1.2 Valeurs constantes . . . . . 9-9
9.4.1.3 Variables de bits . . . . . 9-10
9.4.2 Valeurs discrètes (bits internes, entrées discrètes) . . . . . 9-12
9.4.2.1 Registre de l'état du canal . . . . . 9-12
9.4.2.2 Registre de l'état du canal de vitesse . . . . . 9-13
9.4.3 État de la carte . . . . . 9-14
9.4.3.1 Registre d'état 1 – fonctions de base et avancées . . . . . 9-14
9.4.3.2 Registre d'état 2 – diagnostic et état commun n°1 . . . . . 9-15
9.4.3.3 Registre d'état 3 – diagnostic et état commun n°2 . . . . . 9-17
9.4.4 Mise en correspondance des adresses . . . . . 9-17
9.5 Carte AMC8 . . . . . 9-18
9.5.1 Lecture des registres de maintien . . . . . 9-18
9.5.2 État de la carte . . . . . 9-19
9.5.2.1 Registre d'état 1 de la carte AMC8 – fonctions de base et avancées . . . . . 9-19

**10 CONFIGURATION D'UNE CONNEXION PROFINET (LOGICIEL CPUM VERSION 081 OU ULTÉRIEURE) . . . . . 10-1**

10.1 Introduction . . . . . 10-1
10.2 PROFINET . . . . . 10-1
10.2.1 Protocoles PROFINET . . . . . 10-1
10.2.2 Implémentation de PROFINET sur le VM600 . . . . . 10-2
10.3 Communication PROFINET . . . . . 10-3
10.3.1 Traitement des erreurs . . . . . 10-3
10.3.1.1 Erreurs de communication internes . . . . . 10-3
10.3.1.2 Erreurs de communication externes . . . . . 10-4
10.3.1.3 Erreurs système . . . . . 10-4
10.4 PROFINET IO . . . . . 10-4
10.5 Fichiers GSDML . . . . . 10-5
10.6 Fichiers de configuration PROFINET . . . . . 10-6
10.6.1 Cartes CPUM exécutant la version 081 du microprogramme . . . . . 10-6
10.6.1.1 Fichier modbusDefault.cfg . . . . . 10-6
10.6.1.2 Fichier profinet.cfg . . . . . 10-8
10.7 Configuration d'un système VM600 pour fonctionner en tant qu'équipement E/S PROFINET IO . . . . . 10-9
10.7.1 Équipement utilisé . . . . . 10-9
10.7.1.1 Matériel . . . . . 10-9
10.7.1.2 Logiciel . . . . . 10-9
10.7.2 Conditions préalables . . . . . 10-10
10.7.3 Procédure de configuration d'un réseau PROFINET avec SIMATIC STEP 7 . . . . . 10-10
10.7.3.1 Démarrer le logiciel SIMATIC et créer un nouveau projet . . . 10-10
10.7.3.2 Ajouter des objets à un projet . . . . . 10-11
10.7.3.3 Configurer l'automate SIMATIC CPU . . . . . 10-11
10.7.3.4 Ajouter un équipement IO vibro-meter sur SIMATIC STEP 7 . . . . . 10-14
10.7.3.5 Ajouter un rack VM600 au projet . . . . . 10-16
10.7.3.6 Ajouter une carte MPC4 ou AMC8 au projet . . . . . 10-17
10.7.3.7 Contrôler les propriétés d'un objet SIMATIC . . . . . 10-18
10.7.3.8 Configurer l'interface PG/PC . . . . . 10-18
10.7.3.9 Télécharger la configuration vers SIMATIC CPU . . . . . 10-19

10.7.4 Procédures pour utiliser un réseau PROFINET avec SIMATIC STEP 7 . . . . . 10-20
10.7.4.1 Accès à l'automate . . . . . 10-20
10.7.4.2 Surveiller des équipements . . . . . 10-20
10.7.4.3 Affecter des valeurs d'équipements . . . . . 10-21
10.7.4.4 Créer des tables de variables . . . . . 10-22
10.7.4.5 Vérifier et affecter des noms d'équipements . . . . . 10-23

**11 ÉLIMINATION DES PRODUITS EN FIN DE VIE . . . . . 11-1**

**12 ASSISTANCE CLIENT . . . . . 12-1**

12.1 Pour nous contacter . . . . . 12-1
12.2 Assistance technique . . . . . 12-1
12.3 Assistance commerciale et réparations . . . . . 12-2
12.4 Commentaires des clients . . . . . 12-2
Formulaire de retour d'un produit énergétique . . . . . 12-4
Formulaire de commentaires client d'un produit énergétique . . . . . 12-7

**A ANNEXE A : DÉFINITIONS DE REGISTRE MODBUS MPC4 . . . . . A-1**

**B ANNEXE B : DÉFINITIONS DE REGISTRE MODBUS AMC8 . . . . . B-1**

**C ANNEXE C : DÉFINITION DES LOGEMENTS PROFINET MPC4 . . . . . C-1**

**D ANNEXE D : DÉFINITIONS DES LOGEMENTS PROFINET AMC8 . . . . . D-1**

**E ANNEXE E : NOMS DES SYMBOLES . . . . . E-1**

# 1 INTRODUCTION ET PRÉSENTATION DE LA MISE EN RÉSEAU DU VM600

## 1.1 Possibilités de communication

Il existe plusieurs possibilités de communication avec un rack VM600. Elles sont répertoriées ci-dessous et récapitulées dans la Figure 1-1.

**NOTE :** Les nombres entre parenthèses (...) font référence aux éléments de la Figure 1-1.

- **Ethernet / TCP/IP** (1a et 1b)
  Permet la configuration d'un rack à l'aide du logiciel VM600 MPS (MPS1 et MPS2) ou du logiciel VM600 CMS, de la gamme de produits Meggitt vibro-meter®.

- **Ethernet / CPUM Configurator** (1c)
  Permet la communication avec un rack afin de configurer les cartes réseau (adressage IP) et de modifier les configurations telles que Modbus, NTP et/ou XMx16.
  La communication est prise en charge par une carte CPUM.
  Voir la section 5 Communications Ethernet avec un rack VM600 et la section 6 CPUM Configurator pour plus d'informations.

- **Ethernet / Émulation de terminal** (1d)
  Permet la communication avec un rack afin de configurer les cartes réseau (adressage IP) et de modifier les configurations telles que Modbus, NTP et/ou XMx16.
  La communication est prise en charge par une carte CPUM.
  Voir la section 5 Communications Ethernet avec un rack VM600 et la section 7 Émulation de terminal pour plus d'informations.

**NOTE :** Voir la section 4 Configuration d'une connexion Ethernet pour plus d'informations.

- **Ethernet / Modbus TCP** (2)
  Permet à un système de commande distribué (DCS) de lire des registres et des entrées discrètes.
  La communication est prise en charge par une carte IOCN.
  Voir la section 8 Configuration d'une connexion Modbus (logiciel CPUM version 067 ou antérieure) et la section 9 Configuration d'une connexion Modbus (logiciel CPUM version 071 ou ultérieure) pour plus d'informations.

- **Ethernet industriel / PROFINET** (3)
  Permet à un système de commande distribué (DCS) de lire des registres et des entrées discrètes.
  La communication est prise en charge par une carte IOCN.
  Voir la section 8 Configuration d'une connexion Modbus (logiciel CPUM version 067 ou antérieure) pour plus d'informations.

- **RS-232, RS-485/422 / Modbus RTU** (4)

  Permet à un système de commande distribué (DCS) de lire des registres et des entrées discrètes.

  La communication est prise en charge par une carte IOCN.

  Voir la section 8 Configuration d'une connexion Modbus (logiciel CPUM version 067 ou antérieure) et la section 9 Configuration d'une connexion Modbus (logiciel CPUM version 071 ou ultérieure) pour plus d'informations.

- **RS-232 / Émulation de terminal** (5)

  Permet la communication avec un rack afin de configurer les cartes réseau (adressage IP) et de modifier les configurations telles que Modbus, NTP et/ou XMx16.

  La communication est prise en charge par une carte CPUM.

  Voir la section 3 Communications série avec un rack VM600 et la section 7 Émulation de terminal pour plus d'informations

- **RS-232 / Protocole propriétaire**(6)

  Permet la configuration d'un rack autonome à l'aide du logiciel MPS1, de la gamme de produits Meggitt vibro-meter®.

  La communication est prise en charge directement par une carte MPC4 ou AMC8. Ceci ne rentre pas exactement dans le cadre de la « mise en réseau » du VM600 et ne fera l'objet d'aucune information supplémentaire dans ce manuel.

**NOTE :** Reportez-vous au *Manuel du logiciel des systèmes de protection des machines d'un logiciel de configuration VM600 MPS1* pour plus d'informations.

- **RS-232 / PPP (protocole point à point)**

  Permet la configuration d'un rack à l'aide du logiciel VM600 MPS (MPS1 et MPS2) ou du logiciel CMS, de la gamme de produits Meggitt vibro-meter®.

  La communication est prise en charge par une carte CPUM.

  L'utilisation du protocole point à point (PPP) pour la mise en réseau du VM600 est obsolète et n'est plus prise en charge depuis la version 072 du microprogramme de la carte CPUM.

**NOTE :** Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations.

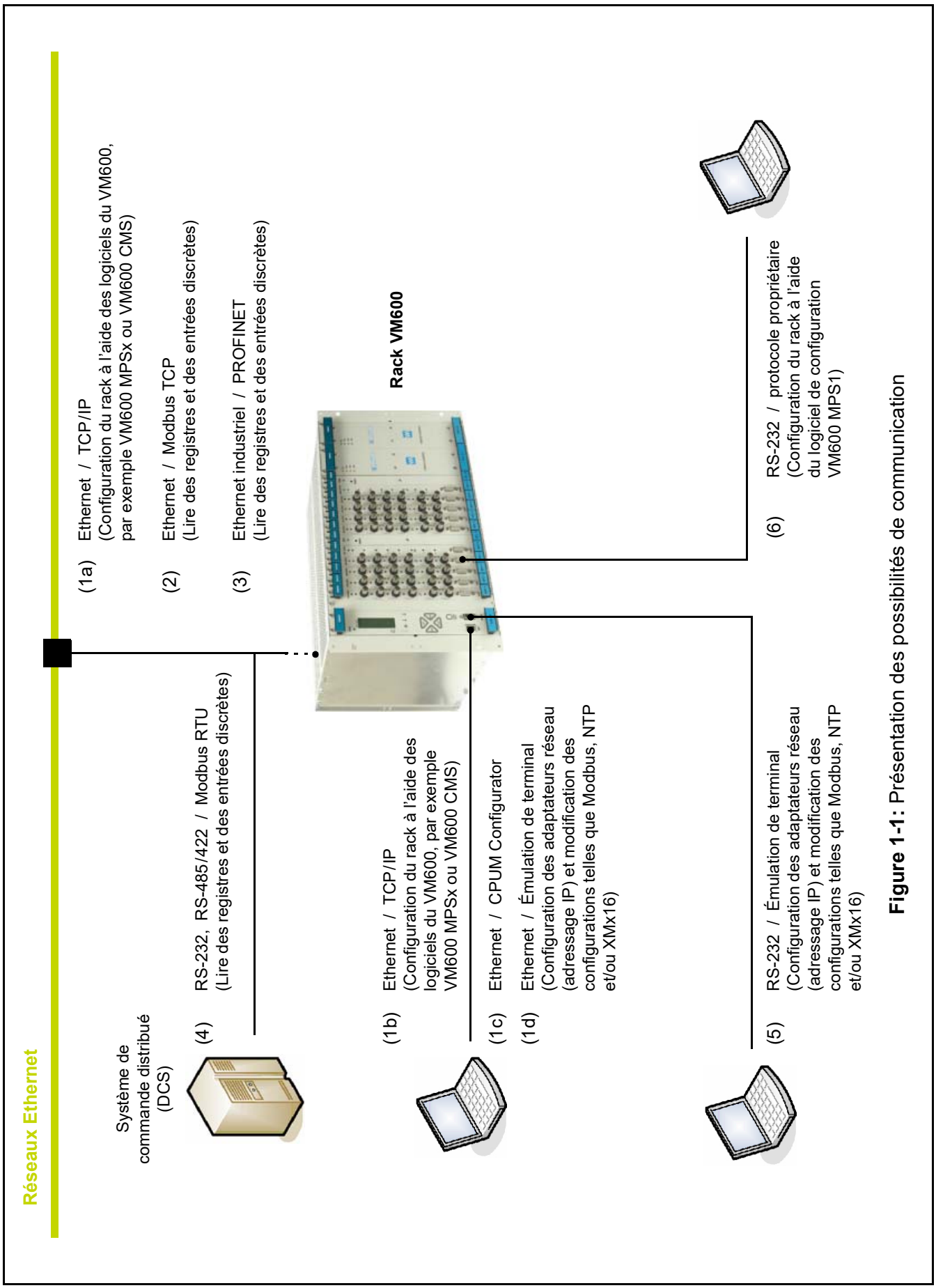



**Figure 1-1:** Présentation des possibilités de communication

## 1.2 Recommandations pour la communication

Comme indiqué dans la section 1.1 Possibilités de communication, il existe plusieurs possibilités de communication avec un rack VM600. Elles ne sont cependant pas toutes prises en charge par toutes les cartes CPUM , comme indiqué dans le Tableau 1-1.

**Table 1-1:** Possibilités de communication et de configuration avec une carte CPUM

| Possibilités de communication | Version du microprogramme de la carte CPUM | | | |
|---|---|---|---|---|
| | 072 ou antérieure | 073 | 074 | 075 ou ultérieure |
| **Prise en charge par la carte CPUM (c'est-à-dire incluse dans le microprogramme CPUM)** | | | | |
| Série (RS-232) | ✓ | ✓ | ✓ | ✓ |
| PPP (RS-232) | ✓ | | | |
| Telnet (TCP/IP) | ✓ | ✓ | ✓ | ✓ |
| Gestionnaire de fichiers Midnight Commander | | | ✓ | ✓ |
| Éditeur de texte vi | ✓ | ✓ | ✓ | ✓ |
| Fichiers de configuration et structure de fichiers | Dépend de la version du microprogramme exécuté sur la carte CPUM. Voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM. | | | |
| **Prise en charge par la carte CPUM (c'est-à-dire compatible avec le microprogramme CPUM)** | | | | |
| CPUM Configurator | | | | ✓ |

En règle générale, les pratiques préférentielles recommandées pour travailler avec les racks VM600 sont les suivantes :

- **Rack VM600 contenant une carte CPUM exécutant la version 075 ou ultérieure du microprogramme**

  Si la connexion avec le rack VM600 est établie via Ethernet (TCP/IP), utilisez le logiciel CPUM Configurator pour travailler avec la carte CPUM et configurer le rack.

  Si le logiciel CPUM Configurator n'est pas disponible, établissez une connexion Telnet avec le rack VM600 au moyen du programme d'émulation de terminal PuTTY et utilisez le gestionnaire de fichiers Midnight Commander pour configurer la carte CPUM.

  Une connexion FTP peut également être établie avec le rack VM600 au moyen du client FileZilla afin de télécharger les fichiers de configuration afin qu'ils puissent être modifiés localement sur un ordinateur, puis chargés sur la carte CPUM.

  Vous pouvez également établir une connexion Telnet avec le rack VM600 au moyen de n'importe quel émulateur de terminal et utiliser l'éditeur de texte vi pour configurer la carte CPUM, comme indiqué dans le *Manuel de mise en réseau du VM600, édition 6 (ou antérieure)*.

- **Rack VM600 contenant une carte CPUM exécutant la version 074 ou antérieure du microprogramme**

  Si la connexion avec le rack VM600 est établie via Ethernet (TCP/IP), établissez une connexion Telnet avec le rack VM600 au moyen du programme d'émulation de terminal PuTTY et utilisez le gestionnaire de fichiers Midnight Commander pour travailler avec la carte CPUM et configurer le rack.

Une connexion FTP peut également être établie avec le rack VM600 au moyen du client FileZilla afin de télécharger les fichiers de configuration afin qu'ils puissent être modifiés localement sur l'ordinateur hôte, puis chargés sur la carte CPUM.

Vous pouvez également établir une connexion Telnet avec le rack VM600 au moyen de n'importe quel émulateur de terminal et utiliser l'éditeur de texte vi pour configurer la carte CPUM, comme indiqué dans le *Manuel de mise en réseau du VM600, édition 6 (ou antérieure)*.

Pour utiliser le logiciel CPUM Configurator ou une connexion Telnet à un rack VM600, l'adresse IP de la carte CPUM doit être connue.
Si l'adresse IP de la carte CPUM n'est pas connue, une connexion série (RS-232) à la carte peut être utilisée pour obtenir l'adresse IP de la carte. Voir la section 3.4 Découverte de l'adresse IP d'une carte CPUM.

## 1.3 Logiciels et protocoles de communication

Cette section fournit un bref aperçu des logiciels et des protocoles pouvant être utilisés pour configurer et gérer des racks VM600, notamment :

- Pour un rack VM600 contenant une carte CPUM exécutant la version 075 ou ultérieure du microprogramme, CPUM Configurator est l'outil recommandé.
- Pour un rack VM600 contenant une carte CPUM exécutant la version 074 du microprogramme, PuTTY, Midnight Commander et/ou le client FileZilla sont les outils recommandés.
- Pour un rack VM600 contenant une carte CPUM exécutant la version 073 ou antérieure du microprogramme, PuTTY, vi et/ou le client FileZilla sont les outils recommandés.

### 1.3.1 CPUM Configurator

CPUM Configurator est un programme qui communique avec une carte CPUM dans un rack VM600 via une liaison Ethernet (TCP/IP). Fondamentalement, il s'agit d'un logiciel qui fournit une interface utilisateur graphique pour une session Telnet entre le logiciel CPUM Configurator et une carte CPUM (serveur Telnet), et qui s'utilise principalement pour la configuration des cartes CPUM et la gestion des racks VM600 sur les liaisons Ethernet.

Le logiciel CPUM Configurator est inclus avec le logiciel VM600 MPSx version 2.7 ou ultérieure et est copié sur l'ordinateur dans le cadre de la procédure d'installation du logiciel VM600 MPSx. Il peut également être obtenu auprès du service clientèle de Meggitt (voir la section 12.1 Pour nous contacter).

**NOTE :** CPUM Configurator est compatible avec les cartes CPUM exécutant la version 075 ou ultérieure du microprogramme (voir le Tableau 1-1).

Pour utiliser le logiciel CPUM Configurator pour configurer un rack VM600, l'adresse IP de la carte CPUM doit être connue.

### 1.3.2 Telnet

Telnet est un protocole réseau utilisé sur Internet ou un réseau local pour fournir des communications bidirectionnelles à l'aide d'une connexion par terminal virtuel. C'est un protocole client-serveur généralement utilisé pour établir une connexion au numéro de port TCP 23.

Telnet s'utilise généralement pour fournir un accès à une interface de ligne de commande (généralement, d'un système d'exploitation) sur un hôte distant. La plupart des équipements réseau et des systèmes d'exploitation équipés d'une pile TCP/IP prennent en charge un service Telnet pour la configuration à distance.

Du point de vue de la mise en réseau du VM600, les programmes d'émulation de terminal tels que PuTTY ou HyperTerminal sont des clients Telnet qui permettent la connexion au serveur Telnet en cours d'exécution (« écoute ») sur la carte CPUM.

**NOTE :** HyperTerminal est le programme d'émulation de terminal fourni avec Microsoft® Windows® XP ou version antérieure. Cependant, HyperTerminal est obsolète et n'est plus inclus avec les systèmes d'exploitation Windows tels que Windows Vista, Windows 7 et Windows 8.

#### 1.3.2.1 PuTTY

PuTTY est un programme d'émulation de terminal qui prend en charge les protocoles réseau tels que Telnet. Une session PuTTY peut être configurée en tant que client Telnet afin de communiquer avec une carte CPUM (serveur Telnet).

PuTTY prend également en charge les communications série et peut être configuré pour communiquer directement avec une carte VM600, comme une carte CPUM ou MPC4, via le connecteur RS-232 sur le panneau avant de la carte.

PuTTY est un logiciel gratuit et open-source, disponible à l'adresse :

- http://www.putty.org

Pour utiliser PuTTY pour communiquer avec un rack VM600 sur une liaison Ethernet, l'adresse IP de la carte CPUM dans le rack doit être connue (voir la section 5 Communications Ethernet avec un rack VM600). L'adresse IP peut être obtenue par l'intermédiaire de PuTTY et d'une liaison série (voir la section 3 Communications série avec un rack VM600).

D'autres logiciels d'émulation de terminal tels que HyperTerminal (ou un terminal compatible VT100) peuvent être utilisés à la place de PuTTY.

#### 1.3.2.2 Client FileZilla

Le client FileZilla est un programme client FTP qui prend en charge les protocoles réseau tels que le protocole de transfert de fichiers (FTP) et le protocole de transfert de fichiers sécurisé (SFTP). Une session du client FileZilla est un client FTP qui peut communiquer avec une carte CPUM (serveur FTP).

Le client FileZilla est un logiciel gratuit et open-source, disponible à l'adresse :

- https://filezilla-project.org

Pour utiliser le client FileZilla pour configurer un rack VM600; l'adresse IP de la carte CPUM dans le rack doit être connue.

D'autres logiciels clients FTP peuvent être utilisés à la place du client FileZilla.

#### 1.3.2.3 Notepad++

Notepad ++ est un programme d'éditeur de texte qui prend en charge l'édition de fichiers texte. Contrairement au Bloc-notes, l'éditeur de texte intégré à Windows Notepad ++ prend en charge l'édition par onglets, ce qui permet de travailler avec plusieurs fichiers ouverts dans une seule fenêtre.

Notepad++ est un logiciel gratuit et open-source, disponible à l'adresse :

- http://notepad-plus-plus.org

D'autres programmes d'édition de texte peuvent être utilisés à la place de Notepad++.

### 1.3.3Midnight Commander

Midnight Commander est un programme gestionnaire de fichiers doté d'une interface utilisateur textuelle qui permet d'afficher un système de fichiers et d'effectuer des opérations de gestion de fichiers telles que la sélection, la copie, l'affichage et l'édition.

Midnight Commander est inclus dans la version 075 ou ultérieure du microprogramme de la carte CPUM afin de faciliter le travail sur les fichiers de configuration de la carte CPUM. La commande Midnight Commander (`mc`) peut être utilisée dans une session d'émulation de terminal pour fournir manuellement toutes les opérations qui sont automatiquement prises en charge par CPUMConfigurator.

Comme Midnight Commander est inclus dans le microprogramme exécuté sur la carte CPUM (voir le Tableau 1-1), vi est la seule alternative qui peut être utilisée pour éditer les fichiers de configuration directement sur la carte (voir la section 1.3.4 Éditeur vi). Un client FTP peut cependant être utilisé pour télécharger les fichiers de configuration afin qu'ils puissent être édités localement sur un ordinateur, puis chargés sur la carte CPUM.

### 1.3.4Éditeur vi

vi est un éditeur de texte qui prend en charge l'édition de fichiers texte.

vi est inclus dans toutes les versions du microprogramme CPUM afin de faciliter le travail sur les fichiers de configuration CPUM. La commande vi (`vi`) peut s'utiliser dans une session d'émulation de terminal pour éditer manuellement les fichiers de configuration directement sur une carte CPUM.

Comme vi est inclus dans le microprogramme exécuté sur la carte CPUM (voir le Tableau 1-1), Midnight Commander est la seule alternative qui peut être utilisée pour éditer les fichiers de configuration directement sur la carte (voir la section 1.3.3 Midnight Commander). Un client FTP peut cependant être utilisé pour télécharger les fichiers de configuration afin qu'ils puissent être édités localement sur un ordinateur, puis chargés sur la carte CPUM.

**NOTE :** Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations sur l'éditeur de texte vi.

## 1.4 Quelques termes et définitions de la mise en réseau

Cette section fournit une brève présentation des termes et définitions de la mise en réseau. Elle s'adresse aux utilisateurs ne connaissant pas bien ce domaine.

Des informations supplémentaires sont disponibles dans les documents rédigés sur le sujet, ou sur de nombreux sites Web.

### 1.4.1 Types de réseau

Deux principaux types de réseau sont présentés ci-dessous. Voir aussi la Figure 1-2.

#### 1.4.1.1 Réseaux locaux

Les réseaux locaux (LAN) présentent les caractéristiques suivantes :

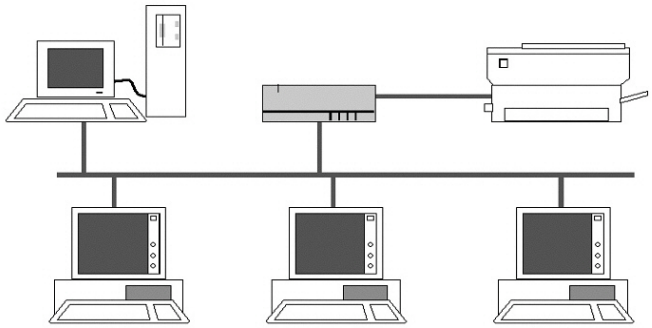
- Ils sont généralement limités à de faibles distances.
- Ils appartiennent aux organisations qui les utilisent.
- Ils utilisent généralement des câbles, même si les LAN sans fil sont de plus en plus fréquents.
- Leur vitesse de transmission se situe dans une fourchette de 10 à 1000 Mbit/s.

### 1.4.1.2 Réseaux étendus

Les réseaux étendus (WAN) présentent les caractéristiques suivantes :

- Ils couvrent de très grandes distances.
- Ils sont généralement utilisés par de grandes entreprises de communication.
- Les systèmes sont souvent connectés à un WAN via des réseaux publics comme le réseau téléphonique.
- Le protocole WAN le plus courant est le protocole TCP/IP.
- Internet est un exemple de WAN.

(a) Réseau local (LAN)



(b) Réseau étendu (WAN)

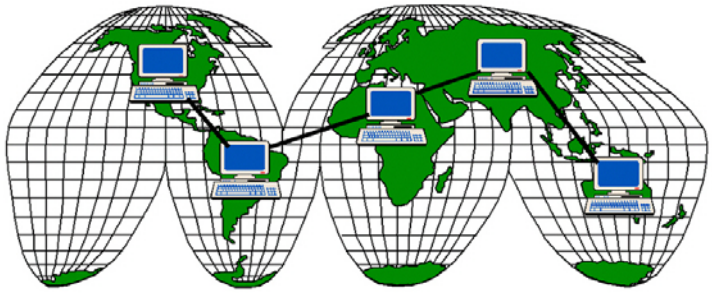


**Figure 1-2:** Types de réseau (LAN et WAN)

### 1.4.2 Connectivité au sein des réseaux

Les réseaux peuvent abriter les périphériques suivants (voir aussi la Figure 1-3) :

**1- Concentrateurs (Hub)**

Un concentrateur est un point de connexion commun aux périphériques d'un réseau. Les concentrateurs sont habituellement utilisés pour connecter les segments d'un LAN. Un concentrateur contient plusieurs ports. Lorsqu'un paquet arrive sur un port, il est copié vers les autres ports de façon à ce que tous les segments du LAN puissent voir tous les paquets.

La disponibilité des commutateurs réseau (switch) à bas prix a largement contribué au déclin des concentrateurs, même si on continue d'en trouver dans les installations anciennes et les applications plus spécialisées.

**2- Commutateurs réseau**

Un commutateur réseau est un périphérique réseau qui connecte les clients, les serveurs et d'autres périphériques réseau entre eux. Également connu sous le nom de commutateur de trames, les commutateurs LAN propriétaires sont courants sur les réseaux Ethernet. Un commutateur à quatre ports est généralement intégré à des routeurs sans fil et filaires pour les petites structures ou les installations à domicile.

**3- Routeurs**

Un routeur est un périphérique réseau qui renvoie les paquets d'un réseau à l'autre. Basés sur des tables de routage internes, les routeurs lisent chaque paquet entrant et décident de la meilleure façon de les renvoyer.

La plupart des routeurs des petites structures et des installations à domicile ne font rien d'autre que d'acheminer les transactions Internet, les emails et autres transmissions électroniques du réseau local au câble ou au modem DSL qui est connecté à un fournisseur de services Internet, puis à Internet.

Dans les grandes entreprises, les routeurs sont également utilisés pour séparer les réseaux locaux (LAN) en sous-réseaux pour des motifs de gestion du réseau. Par exemple, pour équilibrer le trafic dans les groupes de travail ou le filtrer par mesure de sécurité.

**4- Passerelles**

Dans les réseaux informatiques, une passerelle est un nœud sur un réseau qui sert de point d'accès à un autre réseau. Autrement dit, elle fait office d'intermédiaire (point d'entrée/point de sortie) pour les réseaux ou sous-réseaux qui utilisent les mêmes protocoles. (Un routeur peut être considéré comme un type spécial de passerelle.)

La passerelle par défaut est un nœud sur un réseau utilisé lorsqu'une adresse IP ne correspond à aucune autre route de la table de routage. Autrement dit, une passerelle par défaut connecte généralement les réseaux internes et les réseaux externes (par exemple, Internet). Voir la section **5.5 Passerelles** et la section **5.6 Ajout d'une passerelle par défaut à une carte CPUM** pour plus d'informations.

---

**NOTE :** Une passerelle réseau, également appelée passerelle de mise en correspondance//conversion des protocoles, est un périphérique différent qui convertit les paquets d'un protocole vers un autre. En d'autres termes, une passerelle réseau connecte des réseaux qui utilisent des technologies de protocole réseau différentes en effectuant les conversions de protocole requises.

---

**5- Pare-feux**

Un pare-feu autorise ou bloque le trafic entrant et sortant d'un réseau privé ou de l'ordinateur d'un utilisateur. Les pare-feux sont largement utilisés pour donner aux utilisateurs un accès sécurisé à Internet, ainsi que pour séparer le serveur Web public d'une entreprise de son réseau interne. Les pare-feux servent également à assurer la

sécurité de segments internes du réseau. Par exemple, le réseau du service comptabilité peut être vulnérable aux processus de surveillance du trafic mis en place dans une entreprise.
Dans les installations à domicile, un pare-feu personnel est généralement un logiciel installé sur l'ordinateur de l'utilisateur. Dans les grandes entreprises, un pare-feu peut être un logiciel dans un routeur, un serveur ou une machine autonome. Ce peut être un simple routeur qui filtre les paquets non souhaités, ou il peut s'agir d'une combinaison de routeurs et de serveurs effectuant chacun une action de filtrage spécifique.

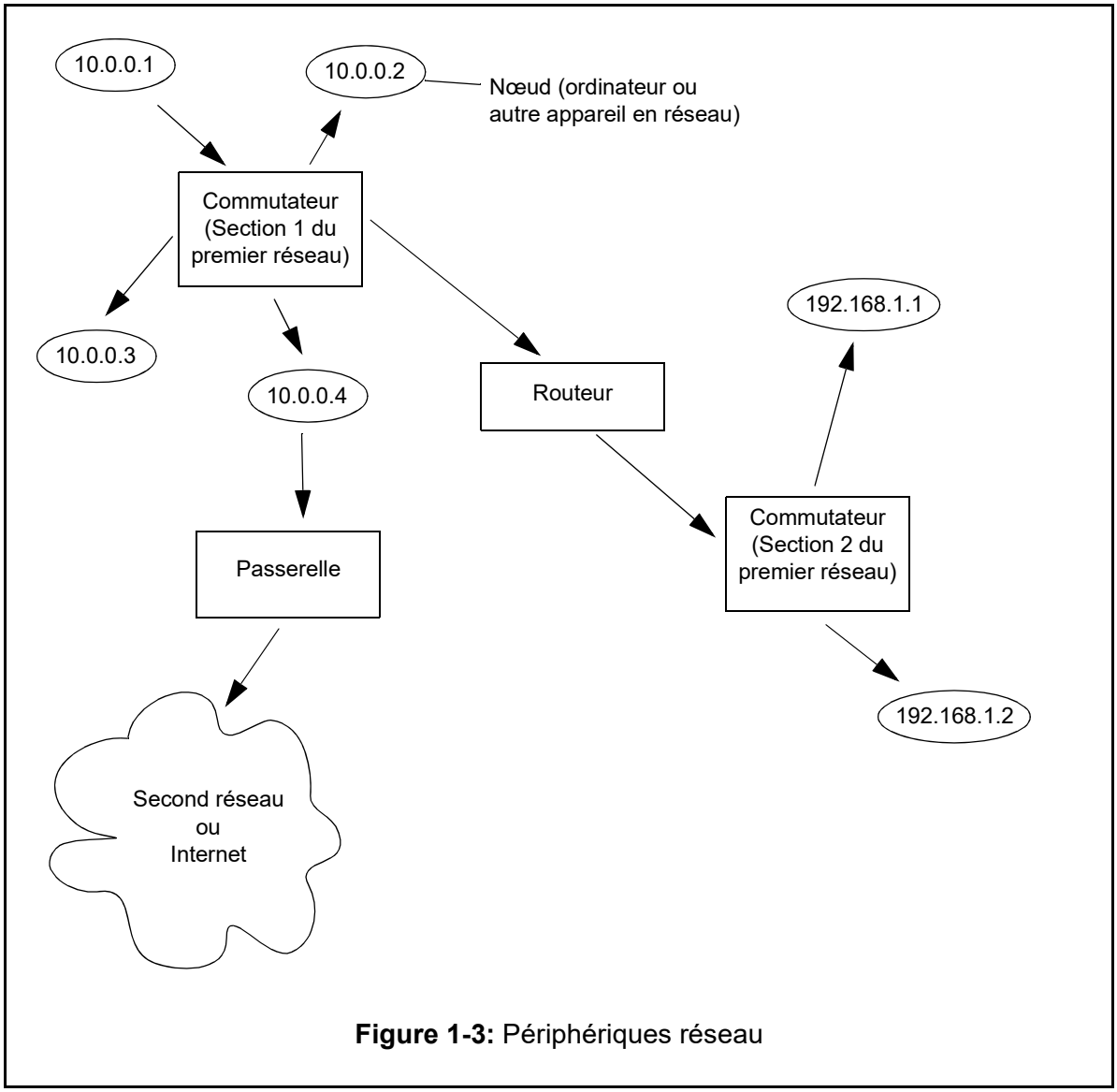



**Figure 1-3:** Périphériques réseau

### 1.4.3 Ethernet

Ethernet fournit une méthode de communication sur large bande passante entre les périphériques connectés à un LAN. Actuellement, il s'agit de la technologie de mise en réseau d'ordinateurs la plus fréquemment utilisée.

La spécification Ethernet d'origine a servi de base à la norme IEEE 802.3. Ceci désigne à la fois le support de transmission physique et la méthode de transmission.

Il existe actuellement plus de 20 variantes d'Ethernet. Les plus applicables au système VM600 sont les suivantes :

**1-** 10BASE-T

- Fonctionne à 10 Mbit/s et utilise des méthodes de transmission en bande de base.
- Utilise le type de câble le moins cher, comme les paires torsadées blindées (STP) ou non blindées (UTP). (Les non blindées sont les plus courantes.)
- La longueur maximum du segment de câble est de 100 mètres.
- Les câbles utilisent des connecteurs modulaires 8P8C (souvent appelés RJ-45).
- Des topologies de type étoile ou bus en étoile sont créées, dans lesquelles les systèmes sont connectés entre eux à l'aide de concentrateurs.
- Ce réseau est simple à étendre.

**2-** 100BASE-T (également appelé Fast Ethernet)

- Fonctionne à 100 Mbit/s et utilise des méthodes de transmission en bande de base.
- Il existe diverses variantes dépendant du support de transmission physique :

  100BASE-TX (2 paires torsadées de grande qualité autorisant une transmission en duplex intégral).

  100BASE-T4 (4 paires torsadées de qualité normale).

  100BASE-FX (2 câbles en fibre optique multimodale).
- Topologie en étoile, laquelle nécessite par conséquent un concentrateur.
- A efficacement remplacé le 10BASE-T.

**3-** 1000BASE-T (également appelé Gigabit Ethernet).

- Fonctionne à 1000 Mbit/s et utilise des méthodes de transmission en bande de base.
- Il existe diverses variantes dépendant du support de transmission physique :

  1000BASE-T (câblage à paires torsadées CAT-5, CAT-5e, CAT-6 ou CAT-7).

  1000BASE-TX (câblage à paires torsadées CAT-6 ou CAT-7).

  1000BASE-LX (câbles à fibre optique multimodale ou monomodale).
- Principalement utilisé pour les câbles principaux des réseaux ou les connexions serveur à serveur (clustering).
- Migration progressive en vue de remplacer le 100BASE-T.

Outre le fait de gérer des topologies de réseaux LAN plus complexes, la communication Ethernet d'un VM600 peut être aussi simple que d'utiliser un câble inverseur pour connecter le rack VM600 directement à l'ordinateur hôte. Voir la section 4 Configuration d'une connexion Ethernet pour de plus amples informations.

### 1.4.4 Protocoles réseau

Les protocoles réseau importants pour le système VM600 sont les suivants :

- TCP/IP.
- DHCP.

#### 1.4.4.1 Protocole TCP/IP

- Le protocole TCP/IP (Transmission Control Protocol / Internet Protocol, protocole de contrôle de transmission / protocole d'Internet) le plus largement utilisé dans le monde (pour Internet).
- Les systèmes d'exploitation réseau (NOS) les plus importants gèrent l'utilisation de TCP/IP.
- Initialement conçu pour les WAN, il est à présent couramment utilisé pour les LAN.
- Protocole routable qui offre une interopérabilité et un interréseautage véritables entre des NOS distincts.
- Le protocole DHCP (Dynamic Host Configuration Protocol, protocole de configuration dynamique des hôtes) permet l'attribution automatique des adresses IP.
- Il fournit un support de routage inter-réseaux complet.

#### 1.4.4.2 Protocole DHCP (dynamic host configuration protocol)

Le protocole DHCP (Dynamic Host Configuration Protocol) est un protocole de configuration réseau pour les réseaux IP qui simplifie l'attribution des adresses IP aux périphériques du réseau.

L'utilisation de DHCP sur un réseau signifie qu'un administrateur réseau n'a pas à attribuer manuellement et que l'utilisateur n'a pas à entrer manuellement une adresse IP pour chaque périphérique du réseau.

**NOTE :** Contactez votre service informatique ou votre administrateur réseau pour savoir si un serveur DHCP est disponible sur votre réseau.

Au lieu de cela, le réseau utilise un serveur DHCP auquel un administrateur réseau a attribué une plage (pool) d'adresses IP disponibles. Lorsqu'un nouveau périphérique exécutant un client DHCP (par exemple, une carte VM600 ou un module DMS (distributed monitoring system) VibroSmart®) rejoint le réseau, le serveur DHCP attribue (loue) une adresse IP au périphérique à partir de son pool. Lorsque le périphérique quitte le réseau, il libère l'adresse IP qui peut alors être réutilisée par le serveur DHCP. Pour ce faire, le serveur DHCP conserve un enregistrement de son pool d'adresses IP – adresses qui sont utilisées et adresses qui peuvent encore être utilisées (et réutilisées).

Il est assez courant, pour un réseau, d'utiliser une combinaison d'adresses réseau attribuées manuellement (statiques) et d'adresses réseau attribuées par DHCP (dynamiques) et pour ses périphériques connectés. Cependant, une bonne gestion du réseau est nécessaire pour s'assurer qu'une adresse IP qui a déjà été allouée à un serveur DHCP n'est pas manuellement attribuée (réutilisée) ailleurs dans le réseau.

**NOTE :** Il est toujours recommandé de contacter votre service informatique ou votre administrateur réseau pour plus d'informations avant d'attribuer manuellement une adresse IP statique à un périphérique de réseau.

### 1.4.5 Adressage IP

#### 1.4.5.1 Règles de base

- Chaque ordinateur d'un réseau doit avoir une adresse hôte unique.
- Chaque réseau doit avoir une adresse réseau unique.
- Chaque adresse IP doit être unique pour le reste du réseau.
- N'utilisez jamais une adresse IP qui ne vous a pas été attribuée si vous vous apprêtez à vous connecter à un réseau, comme un réseau d'entreprise ou Internet.

#### 1.4.5.2 Adressage IP et classes d'adresses

L'adressage IP utilise une adresse 32 bits (c'est-à-dire quatre octets de 8 bits) séparée en deux parties :

- Numéro du réseau
- Numéro de l'hôte.

Chaque octet d'une adresse IP est converti en un nombre décimal compris entre 1 et 255 et les octets de l'adresse sont séparés par des points. Par exemple, 100.100.100.100 est une adresse IP.

Il existe trois classes courantes d'adresses IP (voir également la Figure 1-4) :

- Classe A

  Gère environ 16 millions d'hôtes sur chacun des 127 réseaux.
  Format d'adresse : Réseau.Hôte.Hôte.Hôte

- Classe B

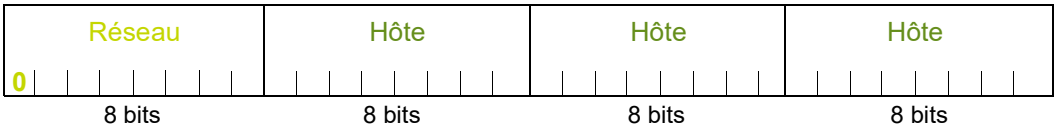
  Gère environ 65000 hôtes sur chacun des quelques 16000 réseaux.
  Format d'adresse : Réseau.Réseau.Hôte.Hôte

- Classe C

  Gère 254 hôtes sur chacun des quelques deux millions de réseaux.
  Format d'adresse : Réseau.Réseau.Réseau.Hôte

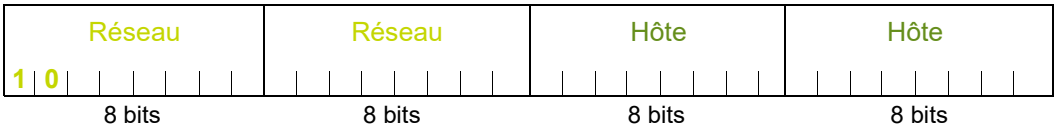
Pour plus d'informations sur les masques de sous-réseau par défaut utilisés par une carte CPUM, voir la section 4.1 Paramètres d'adresse IP par défaut.

### Classe A



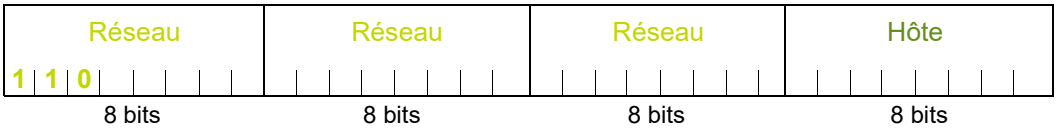

L'adresse réseau binaire commence par « 0 », de sorte que le premier octet peut être compris entre 1 et 127 (décimal).
Le premier octet identifie le réseau et les trois octets restants identifient l'hôte.
Exemple: 104.122.245.10 (où « 104 » identifie le réseau et « 122.245.10 » l'hôte),
Nombre de réseaux possibles : 127,

### Classe B




L'adresse réseau binaire commence par « 10 », de sorte que le premier octet peut être compris entre 128 et 191 (décimal).
Les deux premiers octets identifient le réseau et les deux octets restants identifient l'hôte.
Exemple: 155.113.16.102 (où « 155.113 » identifie le réseau et « 16.102 » l'hôte),
Nombre de réseaux possibles : 16384,

### Classe C




L'adresse réseau binaire commence par « 110 », de sorte que le premier octet peut être compris entre 192 et 223 (décimal).
Les trois premiers octets identifient le réseau et l'octet restant identifie l'hôte.
Exemple: 210.222.126.55 (où « 210.222.126 » identifie le réseau et « 55 » l'hôte),
Nombre de réseaux possibles : 2097152,
Nombre maximum d'hôtes : 254.

Note : Bien qu'il soit théoriquement possible d'obtenir 256 adresses hôte dans cet exemple, les adresses 0 (0b00000000 en binaire) et 255 (0b11111111 en binaire) sont réservées.

**Figure 1-4:** Présentation des classes d'adresses IP

### 1.4.5.3 Sous-réseaux et masques de sous-réseaux

Un sous-réseau est une subdivision logiquement visible d'un réseau IP. Les sous-réseaux vous permettent de subdiviser un réseau en unités plus petites et plus faciles à gérer. Avantages de la subdivision en sous-réseaux :

- Réduction du trafic réseau en sélectionnant le chemin utilisé pour envoyer les informations.
- Optimisation des performances du réseau.
- Gestion simplifiée.
- Couverture améliorée sur de longues distances géographiques.

Pour utiliser un sous-réseau, vous devez fournir un masque de sous-réseau qui détermine la taille d'un sous-réseau et les points de terminaison sur le sous-réseau. La raison pour laquelle on parle de masque de sous-réseau est que les bits de l'hôte sont littéralement masqués pour ne laisser que l'adresse de réseau qui débute le sous-réseau. Lorsque vous connaissez le début du sous-réseau et sa taille, vous pouvez déterminer la fin du sous-réseau, ou adresse de diffusion.

En raison du masque de réseau, l'adresse du premier sous-réseau est paire et l'adresse du dernier sous-réseau est impaire. Ces adresses sont des adresses spéciales à des fins spécifiques. Elles ne peuvent pas être utilisées pour l'adressage général des nœuds :

- L'adresse du premier sous-réseau est l'adresse réseau. Elle s'utilise pour identifier le réseau, c'est-à-dire qu'elle sert de désignation officielle pour un sous-réseau particulier.
- L'adresse du dernier sous-réseau est l'adresse de diffusion. Elle s'utilise pour adresser tous les nœuds du réseau en même temps, c'est-à-dire pour l'adressage de diffusion.

Les masques de sous-réseau par défaut (ou masques naturels) sont :

- Classe A : 255.0.0.0 (décimal)
  11111111.00000000.00000000.00000000 (binaire)
  F F 0 0 0 0 0 0 (hexadécimal)

- Classe B : 255.255.0.0
  11111111.11111111.00000000.00000000
  F F F F 0 0 0 0

- Classe C : 255.255.255.0
  11111111.11111111.11111111.00000000
  F F F F F F 0 0

Pour plus d'informations sur les masques de sous-réseau par défaut utilisés par une carte CPUM, voir la section 4.1 Paramètres d'adresse IP par défaut.

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# 2 FICHIERS DE CONFIGURATION ET STRUCTURE DES RÉPERTOIRES DE LA CARTE CPUM

Pour un rack VM600 en réseau, la carte CPUM contient un certain nombre de fichiers différents qui définissent la configuration et le fonctionnement de la carte CPUM et du rack VM600.

Elle inclut notamment un fichier qui fournit des informations de version de base ainsi que des fichiers de configuration pour toutes les interfaces de communication et tous les protocoles pris en charge par une carte CPUM (et la carte IOCN qui lui est associée (en option)), comme Ethernet, série, Modbus, PROFIBUS et PROFINET.

La structure des répertoires et les fichiers de configuration utilisés, tels que `net.cfg` ou `hosts`, dépendent de la version du microprogramme exécuté sur une carte CPUM.

## 2.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme

Pour les cartes CPUM qui exécutent la version 075 ou ultérieure du microprogramme, la structure des répertoires et les fichiers de configuration ont été améliorés et réorganisés. Ils sont aujourd'hui relativement différents par rapport aux cartes exécutant la version 074 ou antérieure du microprogramme, comme indiqué dans le Tableau 2-1.

**Table 2-1:** Évolution des fichiers de configuration pour les cartes CPUM

<table>
<tr><th colspan="4">Version du microprogramme de la carte CPUM</th><th rowspan="2">Description<br>(type d'information)</th></tr>
<tr><th>070<br>ou antérieure</th><th>De 071 à 074</th><th>De 075 à 080</th><th>081<br>ou ultérieure</th></tr>
<tr><td>hosts</td><td>hosts</td><td colspan="2">net.cfg</td><td>Adaptateurs réseau (adressage IP) et passerelle par défaut (version 074 ou antérieure)</td></tr>
<tr><td>mbcfg.2<br>mbcfg.3<br>mbcfg.4<br>mbcfg.5<br>mbcfg.tcp</td><td colspan="3">modbusDefault.cfg</td><td>Communications Modbus.<br>Communications PROFINET (version 081 ou ultérieure).</td></tr>
<tr><td colspan="3"></td><td>profinet.cfg</td><td>Communications PROFINET supplémentaires</td></tr>
<tr><td colspan="2">tcpip.1</td><td colspan="2"></td><td>Adaptateurs réseau, sous-réseaux et passerelle par défaut (version 074 ou antérieure)</td></tr>
<tr><td colspan="4">version.txt</td><td>Informations sur la version</td></tr>
<tr><td colspan="2" rowspan="6"></td><td colspan="2">launcher.cfg</td><td>Processus de lancement (pour surveiller tous les autres processus)</td></tr>
<tr><td colspan="2">ntp.conf</td><td>Serveur NTP</td></tr>
<tr><td colspan="2">sysinit</td><td>Passerelle par défaut</td></tr>
<tr><td colspan="2">uc.cfg</td><td>Informations de résumé</td></tr>
<tr><td colspan="2">xmcsrv.cfg</td><td>Cartes XMx16 – résumé</td></tr>
<tr><td colspan="2">xmcnn.xml</td><td>Carte XMx16 – individuelle</td></tr>
</table>

Voir la section 6.1.1 Fichiers de configuration de la carte CPUM pour plus d'informations sur la structure des répertoires et les fichiers de configuration des cartes CPUM exécutant la version 075 ou ultérieure du microprogramme.

## 2.2 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme

Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations sur la structure des répertoires et les fichiers de configuration des cartes CPUM exécutant la version 074 ou antérieure du microprogramme.

## 2.3 Fichiers de configuration Modbus

Il est parfois nécessaire de modifier l'adresse Modbus d'un rack ou divers paramètres de communication tels que le débit en bauds ou la parité. Il convient à cet effet de modifier les fichiers de configuration Modbus stockés sur la carte CPUM.

Les fichiers de configuration à modifier dépendent de l'implémentation de l'interface logicielle de Modbus, laquelle dépend de la version du microprogramme exécuté sur la carte CPUM.

### 2.3.1 Cartes CPUM exécutant la version 071 ou ultérieure du microprogramme

Pour les cartes CPUM exécutant la version 071 ou ultérieure du microprogramme, le serveur Modbus est plus flexible et entièrement configurable au moyen d'un seul fichier. Ce fichier de configuration est répertorié dans le Tableau 2-2 et est stocké sur la carte CPUM.

**Table 2-2:** Fichier de configuration Modbus pour cartes CPUM exécutant la version 071 ou ultérieure du microprogramme

| Fichier de configuration Modbus | Description |
|---|---|
| `modbusDefault.cfg` | Fichier de configuration pour les communications Modbus et PROFINET (version 081 ou ultérieure) |

Note : La structure des répertoires (emplacement) des fichiers de configuration Modbus sur la carte CPUM est la suivante : `etc/cpum/mbsrv/modbusDefault.cfg`

**NOTE :** Pour les cartes CPUM exécutant la version 071 ou ultérieure du microprogramme, seul le fichier de configuration unique répertorié dans le tableau Tableau 2-2 doit être modifié pour modifier les paramètres de communication. Le fichier de configuration à modifier est le même pour toutes les interfaces de communication.

Les paramètres de communication sont définis dans les premières sections du fichier de configuration `modbusDefault.cfg`, intitulées [RTUx], [TCP] et [GLOBAL]. Voir la Figure 2-1 et la section 9.2 Définition des registres pour plus d'informations.

**NOTE :** Pour assurer la compatibilité avec les équipements existants, le fichier de configuration Modbus par défaut fourni avec tous les systèmes livrés est configuré pour prendre en charge le même comportement que les cartes CPUM existantes (c'est-à-dire les cartes CPUM exécutant la version 067 ou antérieure du microprogramme).

```
[RTU1]
ENABLE          = NO        //YES/NO (default NO)
ADDRESS         = 1         //1-247 (default 1)
DEVICE          = SERIAL_A  //SERIAL_A / SERIAL_B / RS /RSFRONT (default RS)
SPEED           = 9600      //1200-115000 (default 19200)
PARITY          = N         //N/O/E (default E for EVEN)
STOP            = 1         //1,2 (default 1 stop bit)
DATABITS        = 8         //7,8 (default 8 data bits)

[RTU2]
ENABLE          = NO        //YES/NO (default NO)
ADDRESS         = 1         //1-247 (default 1)
DEVICE          = SERIAL_B  //SERIAL_A / SERIAL_B / RS /RSFRONT (default RS)
SPEED           = 9600      //1200-115000 (default 19200)
PARITY          = N         //N/O/E (default E for EVEN)
STOP            = 1         //1,2 (default 1 stop bit)
DATABITS        = 8         //7,8 (default 8 data bits)

[RTU3]
ENABLE          = NO        //YES/NO (default NO)
ADDRESS         = 1         //1-247 (default 1)
DEVICE          = RS        //SERIAL_A / SERIAL_B / RS /RSFRONT (default RS)
SPEED           = 9600      //1200-115000 (default 19200)
PARITY          = N         //N/O/E (default E for EVEN)
STOP            = 1         //1,2 (default 1 stop bit)
DATABITS        = 8         //7,8 (default 8 data bits)

[TCP]
ENABLE          = YES       //YES/NO (default NO)
PORT            = 502       //0-65535 (default 502)

[GLOBAL]
DEFAULT_LONG_ORDER    = LL
DEFAULT_FLOAT_ORDER   = FL
IS_FULLY_COMPATIBLE   = NO  // YES/NO (default N)
IS_PROFINET_ACTIVATED = NO
```

**Figure 2-1:** Extrait d'un fichier de configuration `modbusDefault.cfg` type utilisé pour les communications Modbus

### 2.3.2 Cartes CPUM exécutant la version 067 ou antérieure du microprogramme

Pour les cartes CPUM exécutant la version 067 ou antérieure du microprogramme, le serveur Modbus est configuré au moyen de plusieurs fichiers. Ces fichiers de configuration sont répertoriés dans le Tableau 2-3 et sont stockés sur la carte CPUM.

**Table 2-3:** Fichiers de configuration Modbus pour cartes CPUM exécutant la version 067 ou antérieure du microprogramme

| Fichier de configuration Modbus | Description |
|---|---|
| `mbcfg.2` | Fichier de configuration pour la communication qui se produit via le connecteur **RS-232** de la carte CPUM.<br>Il s'agit d'un connecteur de type D à 9 broches. |
| `mbcfg.3` | Fichier de configuration pour la communication qui se produit via le connecteur **RS** de la carte IOCN.<br>Il s'agit d'un connecteur 6P6C (RJ11/RJ25). |
| `mbcfg.4` | Fichier de configuration pour la communication qui se produit via les connecteurs **A** de la carte IOCN.<br>Il s'agit de connecteurs 6P6C (RJ11/RJ25). |
| `mbcfg.5` | Fichier de configuration pour la communication qui se produit via les connecteurs **B** de la carte IOCN.<br>Il s'agit de connecteurs 6P6C (RJ11/RJ25). |
| `mbcfg.tcp` | Fichier de configuration pour la communication qui se produit via le connecteur **1** de la carte IOCN pour l'interface réseau principale (ETH1) et via le connecteur **2** du panneau avant de la carte IOCN pour l'interface réseau secondaire (ETH2), si elle est installée. Il s'agit de connecteurs 8P8C (RJ45). |

**NOTE :** Pour les cartes CPUM exécutant la version 067 ou antérieure du microprogramme, seuls certains des fichiers de configurations répertorié dans le Tableau 2-3 doivent être modifiés pour modifier les paramètres de communication. Le fichier de configuration à modifier dépend de l'interface de communication utilisée.

Reportez-vous au *Manuel du matériel du système de protection des machines (MPS) du VM600* pour plus d'informations sur les connecteurs et les interfaces pris en charge par une paire de cartes CPUM / IOCN.

## 2.4 Fichier de configuration PROFINET

### 2.4.1 Cartes CPUM exécutant la version 081 ou ultérieure du microprogramme

Pour les cartes CPUM exécutant la version 081 ou ultérieure du microprogramme, l'implémentation de l'interface logicielle PROFINET s'appuie sur le serveur Modbus existant en utilisant un fichier de configuration PROFINET supplémentaire. Ce fichier de configuration est répertorié dans le Tableau 2-4 et est stocké sur la carte CPUM.

**Table 2-4:** Fichiers de configuration PROFINET pour cartes CPUM exécutant la version 081 ou ultérieure du microprogramme

| Fichier de configuration PROFINET | Description |
|---|---|
| `profinet.cfg` | Fichier de configuration supplémentaire pour les communications PROFINET (version 081 ou ultérieure) |

Note : La structure des répertoires (emplacement) du fichiers de configuration PROFINET sur la carte CPUM est la suivante : `etc/cpum/mbsrv/profinet.cfg`

Voir la section 10.6 Fichiers de configuration PROFINET pour plus d'informations.

### 2.4.2 Cartes CPUM exécutant la version 801 du microprogramme

Pour les cartes CPUM exécutant la version 801 du microprogramme, reportez-vous au *Manuel de mise en réseau du VM600 , édition 8 (ou antérieure),* pour plus d'informations sur la structure des répertoires et les fichiers de configuration liés à la mise en œuvre antérieure de l'interface logicielle PROFINET.

## 2.5 Utilisation des fichiers de configuration

Les fichiers de configuration d'une carte CPUM peuvent être affichés ou modifiés de différentes façons. Voir aussi la section 1.2 Recommandations pour la communication.

Pour les cartes CPUM exécutant la version 075 ou ultérieure du microprogramme :

- Utiliser une liaison Ethernet, soit pour modifier directement les fichiers sur la carte CPUM, soit pour télécharger les fichiers à partir de la carte CPUM, les modifier localement à l'aide d'un éditeur de texte sur un ordinateur, puis les charger sur la carte CPUM. Utiliser logiciel CPUM Configurator, par exemple (voir la section 6 CPUM Configurator).
- Utiliser une liaison Ethernet pour télécharger les fichiers à partir de la carte CPUM, les modifier localement à l'aide d'un éditeur de texte sur un ordinateur, puis les charger sur la carte CPUM. Par exemple, une session FTP utilisant le Client FileZilla et Notepad++ (un éditeur de texte).
- Utiliser une liaison Ethernet pour afficher et modifier directement les fichiers sur la carte CPUM. Par exemple, une session Telnet exécutant Midnight Commander sur la carte CPUM.

Pour les cartes CPUM exécutant la version 074 du microprogramme :

- Utiliser une liaison Ethernet pour télécharger les fichiers à partir de la carte CPUM, les modifier localement à l'aide d'un éditeur de texte sur un ordinateur, puis les charger sur la carte CPUM. Par exemple, une session FTP utilisant le Client FileZilla et Notepad++ (un éditeur de texte).
- Utiliser une liaison série ou Ethernet pour afficher et modifier les fichiers sur la carte CPUM. Par exemple, une session Telnet exécutant Midnight Commander sur la carte CPUM.

Pour les cartes CPUM exécutant la version 073 ou antérieure du microprogramme :

- Utiliser une liaison série ou Ethernet pour afficher et modifier directement les fichiers sur la carte CPUM. Par exemple, une session série ou Telnet exécutant l'éditeur de texte vi sur la carte CPUM.
- Utiliser une liaison Ethernet pour télécharger les fichiers à partir de la carte CPUM, les modifier localement à l'aide d'un éditeur de texte sur l'ordinateur, puis les charger sur la carte CPUM. Par exemple, une session FTP utilisant le Client FileZilla et Notepad++ (un éditeur de texte).

Pour plus d'informations sur les liaisons Ethernet et les outils utilisés, voir les sections 4 Configuration d'une connexion Ethernet, 5 Communications Ethernet avec un rack VM600, 6 CPUM Configurator et 7 Émulation de terminal. Pour plus d'informations sur les liaisons série et les outils utilisés, voir la section 3 Communications série avec un rack VM600.

En général, les communications série s'utilisent uniquement avec un rack VM600 en réseau afin de découvrir l'adresse IP de la carte CPUM. En effet, lorsque l'adresse IP d'une carte CPUM (rack VM600) est connue, la gestion de racks VM600 en réseau peut être plus facilement réalisée par l'intermédiaire de communications Ethernet telles que :

- Le logiciel CPUM Configurator exécuté sur un ordinateur (voir la section 6 CPUM Configurator).
- Une session d'émulation de terminal sur un ordinateur exécutant Midnight Commander sur la carte CPUM (voir la section 7 Émulation de terminal).

## 2.6 Configuration des cartes CPUM

Les cartes CPUM sont généralement fournies pré-configurées avec la configuration spécifiée au moment de la commande, à partir du numéro de commande (PNR) et d'un code d'option particulier.

**NOTE :** Un rack VM600 est généralement entièrement configuré en usine avant la livraison afin qu'il puisse être utilisé tel quel. Autrement dit, si aucune autre configuration n'est spécifiée, un rack VM600 est fourni avec une configuration Modbus « standard », telle que décrite dans la section 2.6.1 Configurations standard.
Pour plus d'informations sur les autres possibilités de configuration, y compris les configurations spécifiques au client, contactez Meggitt SA.

### 2.6.1 Configurations standard

Si aucune autre configuration n'est spécifiée, une carte CPUM est fournie avec une configuration Modbus « standard » qui comprend :

- Des adresses IP par défaut : 10.10.56.56 pour l'interface réseau principale (ETH1) et 10.10.56.156 pour l'interface réseau secondaire (ETH2).
- Une configuration NTP par défaut : synchronisation locale (c'est-à-dire que le CPUM utilise une horloge interne comme référence temporelle).
- Une configuration Modbus par défaut : mappage de registre standard pour les cartes MPC4 et AMC8 dans les logements 3 à 14 du rack VM600.

**NOTE :** Une liste complète des adresses de départ Modbus, sous forme de feuille de calcul Excel® est disponible auprès de Meggitt sur demande.

- Aucune configuration pour les autres cartes dans le rack VM600.

**NOTE :** La configuration Modbus « standard » inclut la prise en charge de Modbus mais PROFINET n'est pas inclus par défaut. Pour plus d'informations sur les possibilités de configuration PROFINET, voir la section 10 Configuration d'une connexion PROFINET (logiciel CPUM version 081 ou ultérieure).

### 2.6.2 Configurations spécifiques au client

Lorsque spécifié, une carte CPUM est fournie avec une configuration « spécifique au client » qui correspond à la configuration convenue avec Meggitt SA et spécifiée à partir d'un numéro de commande (PNR) et d'un code d'option particulier.

Dans de tels cas, la carte CPUM est généralement pré-chargée avec la configuration « spécifique au client » avant l'expédition.

### 2.6.3 Modification des configurations

Le logiciel CPUM Configurator peut s'utiliser pour télécharger une configuration à partir d'une carte CPUM ou pour charger une configuration sur une carte CPUM (voir la section 6 CPUM Configurator).

Il permet par conséquent de modifier la configuration exécutée sur une carte CPUM en fonction des besoins. Exemple : en téléchargeant, en modifiant, puis en chargeant la configuration ou en chargeant une nouvelle configuration différente.

**NOTE :** Il est fortement recommandé d'utiliser le CPUM Configurator pour effectuer copie de sauvegarde de la configuration en cours d'exécution sur une carte CPUM.

**NOTE :** Une copie de sauvegarde de la configuration exécutée sur une carte CPUM est particulièrement importante si vous souhaitez modifier vous-même la configuration fournie, car elle vous permet de revenir à la copie de sauvegarde si vous rencontrez des problèmes.

**NOTE :** Si nécessaire, une copie d'archive de la configuration préchargée sur une carte CPUM avant l'expédition peut être obtenue auprès de Meggitt SA en contactant le Service clientèle et en précisant les informations spécifiques au client telles que le numéro de commande Meggitt SA, le numéro de commande client et/ou le numéro de commande (PNR) et le code d'option particulier de la carte CPUM. Voir la section 12.1 Pour nous contacter.

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# 3 COMMUNICATIONS SÉRIE AVEC UN RACK VM600

Il existe de nombreuses façons de communiquer avec des racks VM600 et de les configurer.

**NOTE :** Reportez-vous au *Manuel du matériel du système de protection des machines (MPS) du VM600* pour plus d'informations sur les racks en réseau et les racks autonomes.

Historiquement, les racks VM600 étaient gérés à l'aide d'un programme d'émulation de terminal (par exemple, PuTTY ou HyperTerminal) exécuté sur un ordinateur pour établir des communications avec le rack VM600 (carte CPUM), puis en exécutant l'éditeur de texte vi fourni par le microprogramme de la carte CPUM pour modifier les fichiers de configuration si nécessaire.

**NOTE :** Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations sur l'utilisation de HyperTerminal et de l'éditeur de texte vi pour gérer des racks VM600 en réseau.

Cependant, lorsque l'adresse IP d'une carte CPUM (rack VM600) est connue, la gestion de racks VM600 en réseau peut être plus facilement réalisée par l'intermédiaire de communications Ethernet telles que :

- Le logiciel CPUM Configurator exécuté sur un ordinateur (voir la section 6 CPUM Configurator).
- Une session d'émulation de terminal sur un ordinateur exécutant Midnight Commander sur la carte CPUM (voir la section 7 Émulation de terminal).

Cependant, pour que les communications Ethernet puissent être utilisées, l'adresse IP de la carte CPUM dans le rack VM600 en réseau doit être connue. Cette information peut être obtenue directement de la carte CPUM au moyen d'une communication en série, comme indiqué ci-dessous.

## 3.1 Paramètres de communication série par défaut

Les paramètres de communication série par défaut pour une carte CPUM sont indiqués dans le Tableau 3-1.

**Tableau 3-1 :** Paramètres de communication série par défaut pour une carte CPUM

| Débit en bauds | Longueur des caractères (bits de données) | Parité | Bits d'arrêt | Contrôle de flux |
|---|---|---|---|---|
| 57600 | 8 | Aucune | 1 | Aucun |

Tout logiciel d'émulation de terminal utilisé pour communiquer avec une carte CPUM doit être configuré avec les paramètres de communication indiqués dans le Tableau 3-1.

## 3.2 Établissement de communications série avec une carte CPUM

La procédure exacte à suivre dépend du système d'exploitation et des outils logiciels utilisés. La description qui suit utilise l'émulateur de terminal PuTTY sous Windows® 7. Si vous utilisez un système d'exploitation différent ou un émulateur de terminal prenant en charge les protocoles de communication série, vous pouvez quand même suivre cette description à titre indicatif.

En cas de questions ou de problèmes, consultez l'aide de Windows ou contactez votre administrateur système. Si aucune solution ne peut être trouvée, contactez votre représentant Meggitt le plus proche.

**1-** Connectez un câble RS-232 entre le connecteur D.sub (**RS-232**) sur le panneau avant de la carte CPUM et le port série d'un ordinateur. Un câble RS-232 droit doit être utilisé, comme indiqué sur la Figure 3-1.

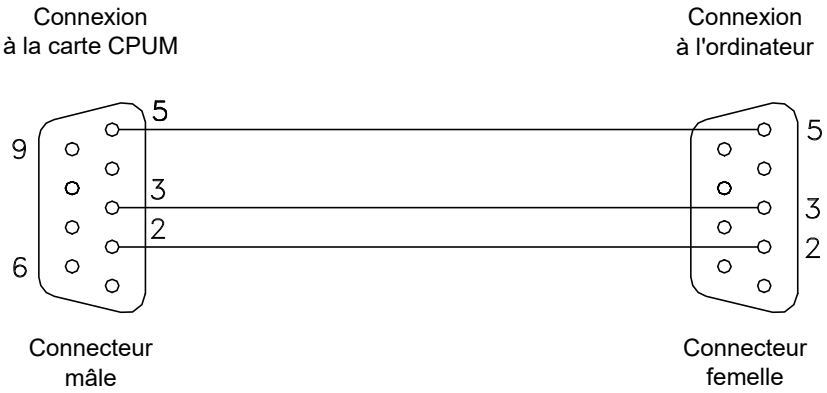



**Figure 3-1:** Câble d'interface utilisé pour connecter une carte CPUM au port série d'un ordinateur exécutant un logiciel d'émulation de terminal

Vous pouvez également utiliser un câble de convertisseur série vers USB (RS-232↔USB) pour un ordinateur sans connecteur RS-232.

**2-** Lancez PuTTY et sélectionnez une connexion **série**, comme indiqué sur la Figure 3-2.

Vous pouvez aussi utiliser un éditeur de terminal différent qui prend en charge les connexions série.

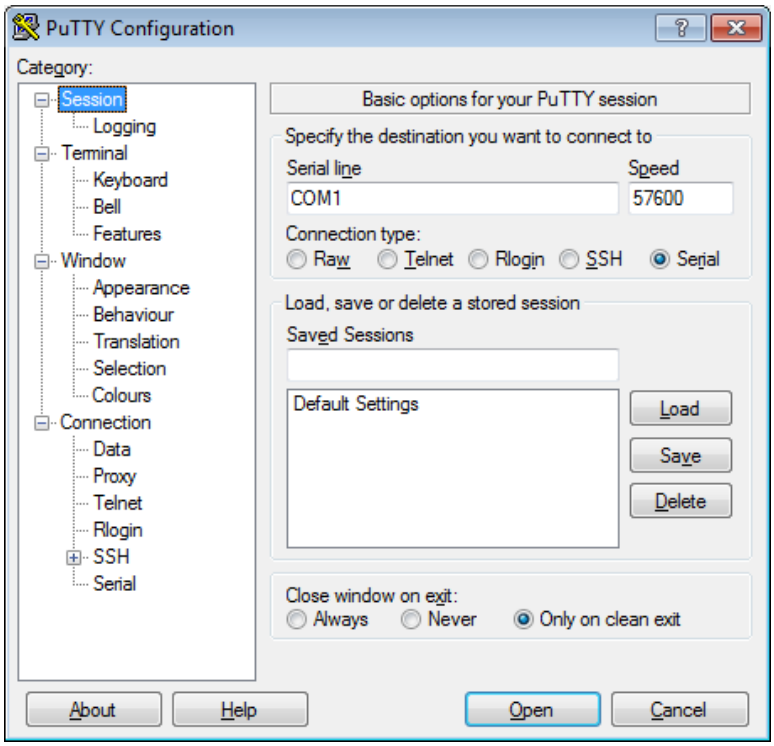



**Figure 3-2:** Utilisation du logiciel d'émulation de terminal PuTTY – sélection d'une connexion série

**3-** Configurez les options de session série PuTTY, comme indiqué sur la Figure 3-3.

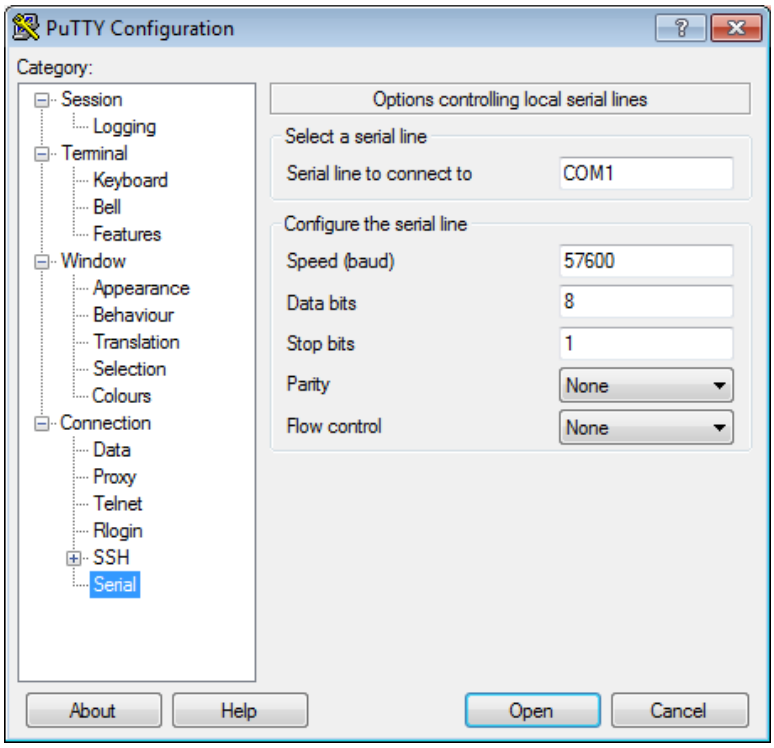



**Figure 3-3:** Utilisation du logiciel d'émulation de terminal PuTTY – configuration d'une connexion série

**4-** Cliquez sur **Open** pour démarrer la session PuTTY, puis appuyez sur la touche ENTRÉE.

**5-** Lorsque vous êtes invité à vous connecter, tapez **user** (identifiant de connexion par défaut), puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
login : user
```

Lorsque vous êtes invité à saisir un mot de passe, tapez **config** (mot de passe par défaut), puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
password : config
```

**6-** Une fois l'identifiant et le mot de passe acceptés et l'utilisateur connecté, l'invite de commande s'affiche. Par exemple :

```
/usr/user #
```

## 3.3 Identification de la version du microprogramme d'une carte CPUM

Il est important de connaître la version du microprogramme exécuté sur une carte CPUM dans la mesure où les fonctions de mise en réseau, la compatibilité logicielle et les fichiers de configuration et la structure peuvent différer, en fonction de la version du microprogramme (voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM).

Cependant, la version du microprogramme est indiquée dans le fichier `version.txt` pour toutes les cartes CPUM.

**1-** Établissez des communications série avec une carte CPUM comme indiqué dans la section 3.2 Établissement de communications série avec une carte CPUM.

**2-** À l'invite de commande ( `/usr/user #` ), tapez **ls**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # ls
```

La commande list dresse la liste des fichiers du répertoire de travail en cours ( `/usr/user` ), y compris le fichier `version.txt` qui contient les informations de version du microprogramme pour une carte CPUM.

**3-** À l'invite de commande ( `/usr/user #` ), tapez **more version.txt**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # more version.txt
```

La commande `more` affiche le contenu du fichier texte spécifié écran par écran. Si nécessaire, appuyez sur ENTRÉE ou ESPACE pour afficher les *n* lignes de texte suivantes.

Vous pouvez également utiliser la commande `cat` pour afficher le contenu d'un fichier sans pause.

### 3.3.1 À propos du fichier version.txt

#### 3.3.1.1 Exemple de fichier version.txt

```
CPU-M : 200-595-075-HHh
CF    : 209-595-300-075

Software version at last startup
--------------------------------
SW_VER : 0.7;5
SW     : 209-595-300-075

Individual SW checksums
-----------------------
xmcsrv           365156814
mbsrv           3104950782
rpcamcsrv       1060545938
rpccmcsrv         49297480
rpccpusrv       1030630127
rpcmpcsrv       1406217733
vmesrv             1903091
testfp          2988981983
vmetest         1047897882
trigwdog        3766606088
setboot         2042170770
launcher        2954264382
```

#### 3.3.1.2 Description

`CPU-M` est le numéro de commande (PNR) de la carte `CPUM`.

Dans ce PNR (`200-595-0SS-xHh`) :

- `SS` représente la version du microprogramme.
- `x` représente la version du matériel :
  - `1` indique la version standard de la carte.
  - `3` indique la version Ethernet redondante de la carte.
  - `4` indique la version série redondante de la carte.
- Les incréments `H` représentent les modifications matérielles majeures qui peuvent affecter l'interchangeabilité du produit.
- Les incréments `h` représentent les modifications matérielles mineures qui n'ont aucune incidence sur l'interchangeabilité du produit.

`CF` correspond au numéro de commande (PNR) de la carte mémoire CompactFlash (contenant le microprogramme) installée sur la carte.

Dans ce PNR (209-595-300-0SS) :

- SS représente la version du microprogramme.

SW_VER correspond à la version du microprogramme exécuté sur la carte. Version 075, par exemple.

SW correspond au numéro de commande (PNR) du microprogramme (intégré à la carte mémoire CompactFlash) installé sur la carte.

**NOTE :** La version du microprogramme exécuté sur la carte CPUM est indiquée à plusieurs endroits. La section 3.3.1 À propos du fichier version.txt, par exemple, utilise la version 075 du microprogramme :

```
CPU-M   : xxx-xxx-075-xxx
CF      : xxx-xxx-xxx-075
SW_VER  : 0.7;5
SW      : xxx-xxx-xxx-075.
```

## 3.4 Découverte de l'adresse IP d'une carte CPUM

### 3.4.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme

Si l'adresse IP d'une carte CPUM n'est pas connue, elle peut malgré tout être découverte par l'intermédiaire d'une connexion série avec le connecteur RS-232 sur le panneau avant de la carte CPUM et d'un logiciel d'émulation de terminal. Pour plus d'informations sur les adresses IP des cartes CPUM, voir la section 4.1 Paramètres d'adresse IP par défaut.

La structure des répertoires et les fichiers de configuration dépendent de la version du microprogramme exécutée sur une carte CPUM (voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM). Pour une carte CPUM qui exécute la version 075 ou ultérieure du microprogramme, les informations d'adresse IP sont définies dans le fichier `net.cfg`.

**1-** Établissez des communications série avec une carte CPUM comme indiqué dans la section 3.2 Établissement de communications série avec une carte CPUM.

**2-** À l'invite de commande ( `/usr/user #` ), tapez **ls**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # ls
```

La commande list dresse la liste des fichiers du répertoire de travail en cours ( `/usr/user` ), y compris le fichier `net.cfg` qui contient les informations d'adresse IP d'une carte CPUM exécutant la version 075 ou ultérieure du microprogramme.

**3-** À l'invite de commande ( `/usr/user #` ), tapez **more net.cfg**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # more net.cfg
```

La commande `more` affiche le contenu du fichier texte spécifié écran par écran. Si nécessaire, appuyez sur ENTRÉE ou ESPACE pour afficher les *n* lignes de texte suivantes.

**4-** Dans le fichier `net.cfg`, les adresses IP des interfaces réseau de la carte CPUM sont affichées sous `#main network connection` sous la forme `net0_ipaddr` (pour l'interface réseau principale (ETH1)) et sous `#redundant network connection` sous la forme `net1_ipaddr` (pour l'interface réseau secondaire (ETH2)).

### 3.4.2 À propos du fichier net.cfg

#### 3.4.2.1 Exemple de fichier net.cfg

```
#VM600 CPUM TCPIP Configuration

#main network connection
net0_ipaddr = 10.10.56.56
net0_netmask = 0xFFFFFF80
net0_force10Mb = 0
#redundant network connection
net1_ipaddr = 10.10.56.156
net1_netmask = 0xFFFFFF80
net1_force10Mb = 1
```

Comme indiqué dans l'exemple de fichier `net.cfg` ci-dessus :

- Le fichier `net.cfg` nécessite des masques de réseau en notation hexadécimale. (Les masques de réseau sont généralement rédigés en notation hexadécimale ou en notation décimale à point (0xFFFFFF80 ou 255.255.255.128), par exemple).
- L'interface réseau principale (ETH1) peut fonctionner à 100 Mbps, paramètre configuré via `net0_force10Mb = 0`.
- L'interface réseau secondaire (ETH2) doit fonctionner à 10 Mbps, paramètre configuré via `net1_force10Mb = 1`.

**NOTE :** Seule l'interface réseau principale (ETH1) peut fonctionner à 100 Mbps : L'interface réseau secondaire (ETH2) est limitée à 10 Mbps, afin d'être rétrocompatible avec les systèmes VM600CPUM les plus anciens.

**NOTE :** Seule l'interface réseau principale (ETH1) peut être utilisée pour les communications PROFINET. L'interface réseau secondaire (ETH2) est limitée à 10 Mbps, tandis que le VM600 CPUM implémente des E/S PROFINET qui nécessitent 100 Mbps.

#### 3.4.2.2 Description du fichier net.cfg

# s'utilise pour les commentaires de ligne.

`#main network connection` est un commentaire utilisé pour indiquer la section du fichier utilisée pour l'interface réseau principale (ETH1) disponible sur le panneau avant de la carte `CPUM` ou sur le panneau avant de la carte `IOCN`.

**NOTE :** L'interface réseau principale (ETH1) d'une carte CPUM peut être acheminée via le connecteur 8P8C (**NET**) qui se trouve sur le panneau avant de la carte CPUM ou via le connecteur 8P8C supérieur (**1**) qui se trouve sur le panneau avant de la carte IOCN associée (en option). Le choix est fait au moyen des cavaliers J42, J40, J52 et J53 de la carte CPUM. Lorsque tous ces cavaliers sont retirés, par exemple, l'interface réseau principale est acheminée via la carte CPUM.
L'interface réseau secondaire (ETH2) d'une carte CPUM est acheminée via le connecteur 8P8C inférieur (**2**) qui se trouve sur le panneau avant de la carte IOCN associée (en option).
Reportez-vous au *Manuel du matériel du système de protection des machines (MPS) du VM600* pour plus d'informations.

`net0_ipaddr` est l'adresse IP utilisée pour l'interface réseau principale (ETH1).

`net0_netmask` est le masque réseau utilisé pour l'interface réseau principale.

`net0_force10Mb` est utilisé pour limiter le débit de transfert de données pour l'interface réseau principale, si nécessaire :

- `net0_force10Mb = 0` autorise un taux de transfert de données pouvant aller jusqu'à 100 Mbps.
- `net0_force10Mb = 1` limite le taux de transfert de données à 10 Mbps, ce qui peut être nécessaire pour la compatibilité avec les réseaux plus anciens.

Ce paramètre peut être nécessaire pour des raisons de compatibilité avec les infrastructures réseau qui utilisent des équipements plus anciens (cartes CPUM).

**NOTE :** Pour une paire de cartes CPUM / IOCN, le contrôleur Ethernet principal (ETH1) peut fonctionner jusqu'à 100 Mbps sur les versions les plus récentes de la carte IOCN (PNR : 200-566-000-113 et versions ultérieures) mais est limité à 10 Mbps sur les versions plus anciennes de la carte IOCN (PNR : 200-566-000-112 et versions antérieures).
Le contrôleur Ethernet secondaire (ETH2) est toujours limité à 10 Mbps.

Sous `#redundant network connection`, `#net1_ipaddr`, `#net1_netmask` et `net1_force10Mb` sont les paramètres équivalents utilisés pour configurer l'interface réseau secondaire (ETH2) disponible sur le panneau avant de la carte `IOCN`, si installée.

### 3.4.3 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme

Pour plus d'informations sur les adresses IP des cartes CPUM, voir la section 4.1 Paramètres d'adresse IP par défaut.

Pour une carte CPUM exécutant la version 074 ou antérieure du microprogramme, les adresses IP sont définies dans le fichier `hosts`. La procédure décrite dans la section 3.4.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme peut par conséquent être utilisée, mais en accédant au fichier `hosts` (plutôt qu'au fichier `net.cfg`).

Par exemple, l'adresse IP de l'interface réseau de la carte CPUM est affichée sous `#Primary Ethernet Port` sous la forme `node1`.

**NOTE :** Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations.

# 4 CONFIGURATION D'UNE CONNEXION ETHERNET

Une connexion Ethernet directe est la méthode de communication la plus rapide et la plus simple avec un rack VM600 (voir la Figure 4-1).

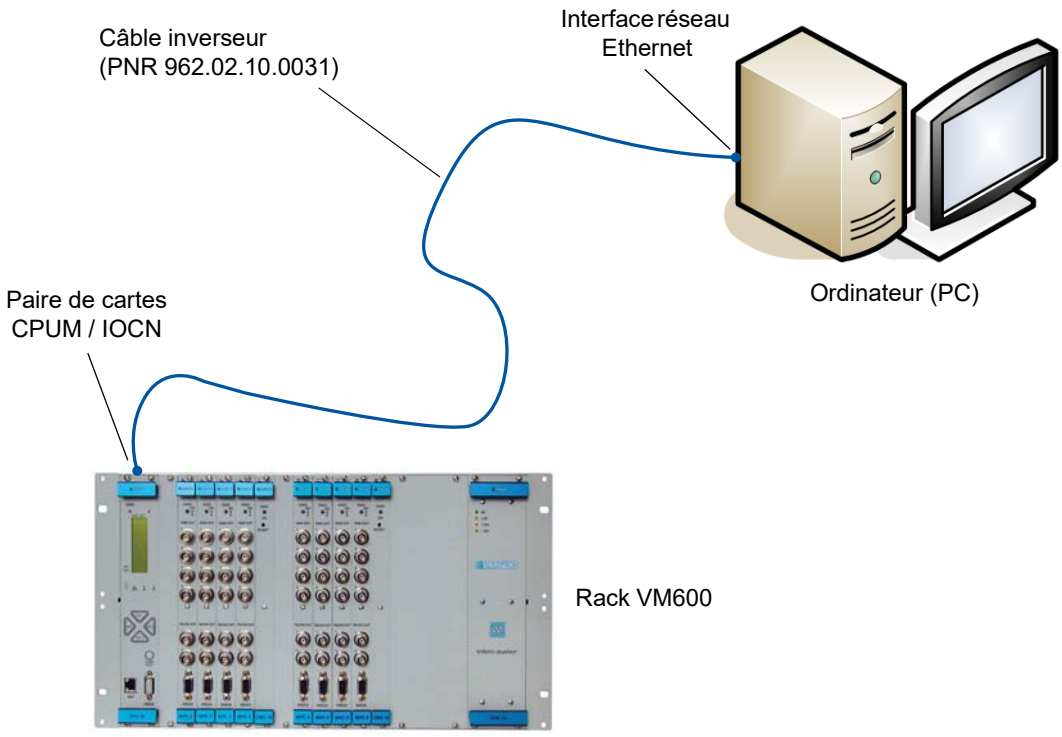



**Figure 4-1:** Connexion directe d'un ordinateur à un rack VM600 (sans utiliser de LAN)

La procédure exacte à suivre dépend du système d'exploitation utilisé. La description suivante utilise l'émulateur de terminal PuTTY sous Windows® 7. Si vous utilisez un système d'exploitation ou un émulateur de terminal différent qui prend en charge les protocoles de communication Telnet, vous pouvez suivre cette description à titre indicatif, mais les boîtes de dialogue et les menus affichés sur les captures d'écran risquent d'être très différents.

En cas de questions ou de problèmes, consultez l'aide de Windows ou contactez votre administrateur système. Si aucune solution ne peut être trouvée, contactez votre représentant Meggitt le plus proche.

## 4.1 Paramètres d'adresse IP par défaut

Comme tout périphérique compatible avec un réseau Ethernet (TCP/IP), une carte CPUM utilise une adresse IP pour qui l'identifie de manière unique sur le réseau.

**NOTE :** Les adresses IP statiques doivent être attribuées et configurées manuellement pour une carte CPUM car celle-ci ne gère pas le protocole DHCP (voir la section 1.4.4.2 Protocole DHCP (dynamic host configuration protocol)).

### 4.1.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme

Les paramètres de communication Ethernet par défaut attribués en usine à une carte CPUM exécutant la version 075 ou ultérieure du microprogramme sont indiqués dans le Tableau 4-1.

**Tableau 4-1 :** Paramètres d'adresse IP par défaut pour une carte CPUM exécutant la version 075 ou ultérieure du microprogramme

| Adresse IP | Masque de réseau | Description |
|---|---|---|
| 10.10.56.56 | 0xFFFFFF80 (255.255.255.128) | Interface réseau principale (ETH1). Il s'agit de l'interface réseau disponible sur la carte CPUM (connecteur **NET**) ou sur la carte IOCN (connecteur **1**). |
| 10.10.56.156 | | Interface réseau secondaire (ETH2). Il s'agit de l'interface réseau disponible sur la carte IOCN (connecteur **2**), si elle est installée. |

**NOTE :** Pour les cartes CPUM exécutant la version 075 du microprogramme, les adresses IP par défaut restent identiques mais le masque réseau par défaut a changé, par rapport à la version 074 ou antérieure du microprogramme (voir le Tableau 4-2).

**NOTE :** L'interface réseau principale (ETH1) d'une carte CPUM peut être acheminée via le connecteur 8P8C (**NET**) qui se trouve sur le panneau avant de la carte CPUM ou via le connecteur 8P8C supérieur (**1**) qui se trouve sur le panneau avant de la carte IOCN associée (en option). Le choix est fait au moyen des cavaliers J42, J40, J52 et J53 de la carte CPUM. Lorsque tous ces cavaliers sont retirés, par exemple, l'interface réseau principale est acheminée via la carte CPUM.
L'interface réseau secondaire (ETH2) d'une carte CPUM est acheminée via le connecteur 8P8C inférieur (**2**) qui se trouve sur le panneau avant de la carte IOCN associée (en option).
Reportez-vous au *Manuel du matériel du système de protection des machines (MPS) du VM600* pour plus d'informations.

Pour une carte CPUM exécutant la version 075 ou ultérieure du microprogramme, les adresses IP sont définies dans le fichier `net.cfg` (voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM). Ces paramètres attribués en usine peuvent être modifiés selon les besoins.

### 4.1.2 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme

Les paramètres de communication Ethernet par défaut attribués en usine à une carte CPUM exécutant la version 074 ou ultérieure du microprogramme sont indiqués dans le Tableau 4-2.

**Tableau 4-2 :** Paramètres d'adresse IP par défaut pour une carte CPUM exécutant la version 074 ou antérieure du microprogramme

| Adresse IP | Masque de réseau | Description |
|---|---|---|
| 10.10.56.56 | 0xFFFF0000 (255.255.0.0) | Interface réseau principale (ETH1). Il s'agit de l'interface réseau disponible sur la carte CPUM (connecteur **NET**) ou sur la carte IOCN (connecteur **1**). |
| 10.10.56.156 | | Interface réseau secondaire (ETH2). Il s'agit de l'interface réseau disponible sur la carte IOCN (connecteur **2**), si elle est installée. |

Pour une carte CPUM exécutant la version 074 ou antérieure du microprogramme, les adresses IP sont définies dans le fichier `hosts` (voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM).

Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations.

## 4.2 Configuration d'un ordinateur pour des communications Ethernet avec un rack VM600

L'adaptateur réseau (carte d'interface réseau) de l'ordinateur à utiliser pour la communication avec un rack VM600 doit être configuré pour être compatible avec l'adresse IP et le masque de sous-réseau utilisés par la carte CPUM, c'est-à-dire que les deux périphériques doivent se trouver dans le même sous-réseau (voir la section 1.4.5 Adressage IP).

**NOTE :** Il est assez courant, pour un réseau, d'utiliser une combinaison d'adresses attribuées par DHCP, attribuées automatiquement par IP (dynamiques) et/ou attribuées manuellement (statiques) pour ses périphériques connectés. Cependant, une bonne gestion du réseau est nécessaire pour s'assurer qu'une adresse IP qui a déjà été allouée d'une manière n'est pas réutilisée ailleurs dans le réseau.
Par conséquent, il est toujours recommandé de contacter votre service informatique ou votre administrateur réseau avant d'attribuer manuellement une adresse IP statique à un rack VM600 (carte CPUM) sur un réseau.

Le fichier `hosts` de l'ordinateur à utiliser pour la communication avec un rack VM600 peut éventuellement être modifié afin que le rack VM600 (carte CPUM) puisse être identifié par un nom d'hôte en plus de l'adresse IP (voir la section 4.2.3 Modification du fichier hosts d'un ordinateur).

### 4.2.1 Configuration requise

**1-** Un ordinateur exécutant l'un des systèmes d'exploitation Windows suivants : Windows 98, Windows NT 4.0, Windows XP ou Windows 7.

**2-** Une adresse IP unique pour votre ordinateur.

**3-** Un rack VM600 contenant une carte CPUM (et la carte IOCN associée (en option)) avec au moins une interface Ethernet.

**4-** Une adresse IP unique pour le rack VM600.

**5-** Un câble croisé Ethernet permettant une communication Ethernet entre l'ordinateur et une carte CPUM (ou IOCN).
Vous pouvez vous procurer un câble croisé Ethernet approprié auprès de Meggitt SA (PNR 962.02.10.0031).

---

**NOTE :** L'adresse IP du rack VM600 (carte CPUM) est paramétrée en usine sur 10.10.56.56 (voir la section 4.1 Paramètres d'adresse IP par défaut). Il s'agit d'une adresse par défaut qu'il est fortement recommandé de modifier.
Pour modifier l'adresse pour une carte CPUM exécutant la version 075 ou ultérieure du microprogramme, modifiez le fichier `net.cfg` stocké sur la carte CPUM (voir la section 3.4.2 À propos du fichier net.cfg).
Pour modifier l'adresse pour une carte CPUM exécutant la version 074 ou antérieure du microprogramme, modifiez le fichier `hosts` stocké sur la carte CPUM (voir la section 3.4.3 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme).

---

### 4.2.2 Configuration de la carte réseau d'un ordinateur

**1-** Affichez les propriétés de la carte réseau sur l'ordinateur. Par exemple :

- Sur un ordinateur Windows 7, dans le menu Start de Windows, cliquez sur **Control Panel > Network and Sharing Center**. Dans la fenêtre Network and Sharing Center qui apparaît, cliquez sur **Change adapter settings** pour afficher la fenêtre Network Connections, puis cliquez sur l'icône Local Area Connection avec le bouton droit et cliquez sur Properties.
- Sur un ordinateur Windows XP, à partir du menu Start de Windows, cliquez sur **Control Panel** > **Network and Internet Connections** > **Network Connections**. Cliquez ensuite sur l'icône Local Area Connection avec le bouton droit et sélectionnez **Properties**.

La boîte de dialogue Local Area Connection Properties s'affiche alors, comme indiqué sur la Figure 4-2.

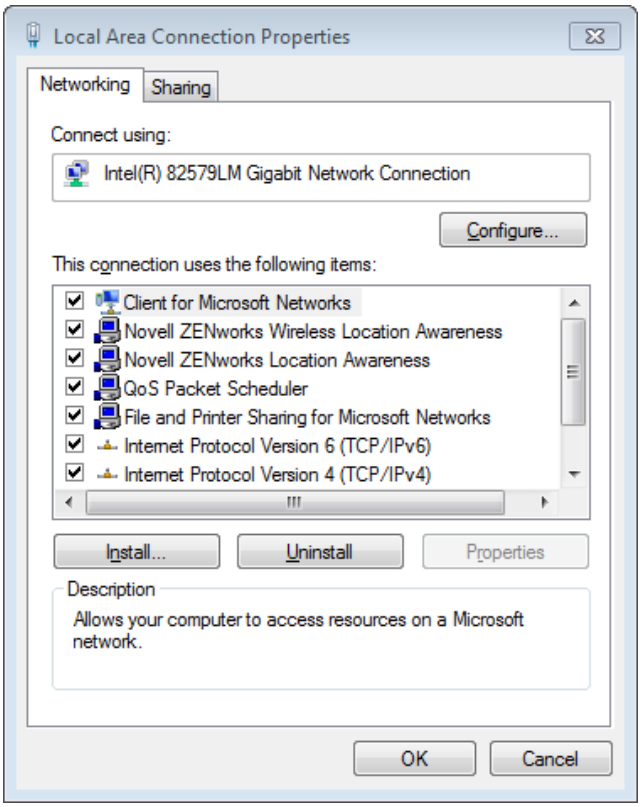



**Figure 4-2:** Boîte de dialogue Local Area Connections de Windows

**2-** Sélectionnez l'élément **Internet Protocol Version 4 (TCP/IPv4)**, puis cliquez sur le bouton **Properties**.

La boîte de dialogue Internet Protocol Version 4 (TCP/IPv4) Properties s'affiche alors, comme indiqué sur la Figure 4-3.

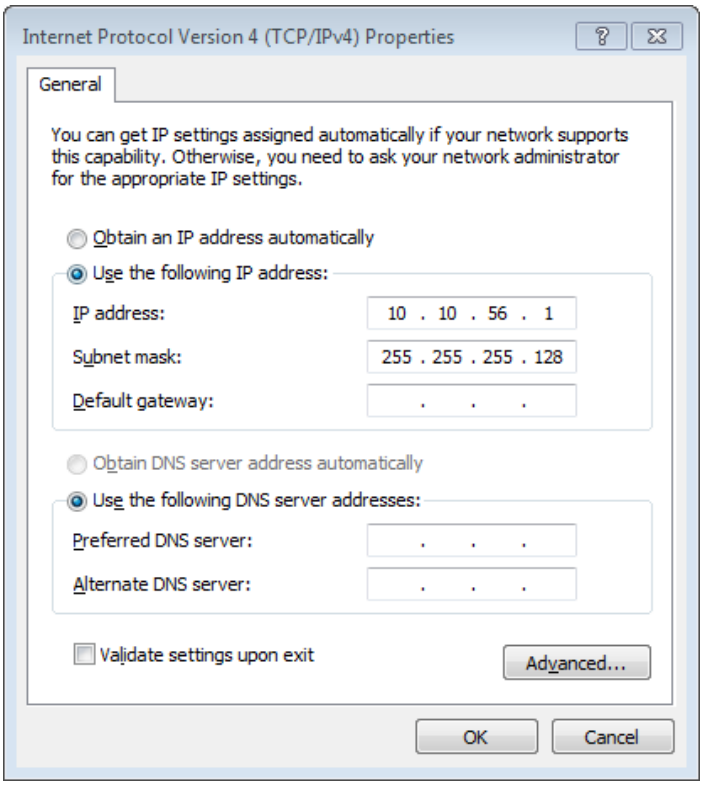



**Figure 4-3:** Boîte de dialogue Internet Protocol Version 4 (TCP/IPv4) Properties

**3-** Sur l'onglet **General**, sélectionnez **Use the following IP address**, puis saisissez les informations suivantes pour cette interface réseau de l'ordinateur :

Sous **IP address**, saisissez l'adresse IP en notation décimale à point.

**NOTE :** Contactez votre service informatique ou votre administrateur réseau pour vous assurer que l'adresse IP statique à utiliser n'est pas déjà utilisée ailleurs sur le réseau.

Sous **Subnet mask**, entrez le masque de sous-réseau en notation décimale à point.

Les masques de réseau peuvent être écrits en notation hexadécimale ou en notation décimale à point (0xFFFFFF80 ou 255.255.255.128, par exemple). La boite de dialogue Windows Internet Protocol Version 4 (TCP/IPv4) Properties exige des masques de réseau en notation décimale à point.

Sous **Default gateway**, entrez la passerelle par défaut en notation décimale à point, le cas échéant.

Cliquez sur **OK** pour fermer la boîte de dialogue Windows Internet Protocol Version 4 (TCP/IPv4), puis cliquez sur **Close** pour fermer la boîte de dialogue Windows Local Area Connections et appliquer les paramètres d'adresse IP à la carte réseau.

### 4.2.3 Modification du fichier hosts d'un ordinateur

Un nom d'hôte est un libellé attribué à un périphérique connecté à un réseau informatique afin d'identifier ce périphérique. Le fichier `hosts` de l'ordinateur s'utilise pour associer l'adresse IP d'un périphérique à un nom d'hôte, de sorte que lorsque le nom d'hôte est utilisé,

l'ordinateur peut se référer au fichier `hosts` et résoudre le nom d'hôte en fonction de son adresse IP.

L'utilisation de nom d'hôtes est facultative, mais elle facilite la gestion des systèmes. Elle facilite l'accès à un périphérique réseau, comparée à la mémorisation et à l'utilisation d'adresses IP, par exemple. Le mappage des adresses IP aux noms d'hôtes contribue également à la résolution des problèmes DNS.

**1-** Ouvrez le fichier `hosts` (local) sur le disque dur de l'ordinateur au moyen d'un éditeur de texte tel que Notepad ++ ou un éditeur similaire. Le fichier `hosts` se trouve dans la zone du système d'exploitation Windows de l'ordinateur. Par exemple : si Windows XP est le système d'exploitation utilisé, le fichier se trouve aux emplacements suivants :

Sur un ordinateur Windows XP, l'emplacement par défaut du fichier `hosts` est `C:\WINNT\system32\drivers\etc`

Sur un ordinateur Windows 7, l'emplacement par défaut du fichier `hosts` est `C:\Windows\System32\drivers\etc`

**2-** Comme indiqué sur la Figure 4-4, le fichier `hosts` répertorie l'adresse IP d'un périphérique suivie de son nom d'hôte, séparés par un ou plusieurs caractères d'espacement, sur une seule ligne.

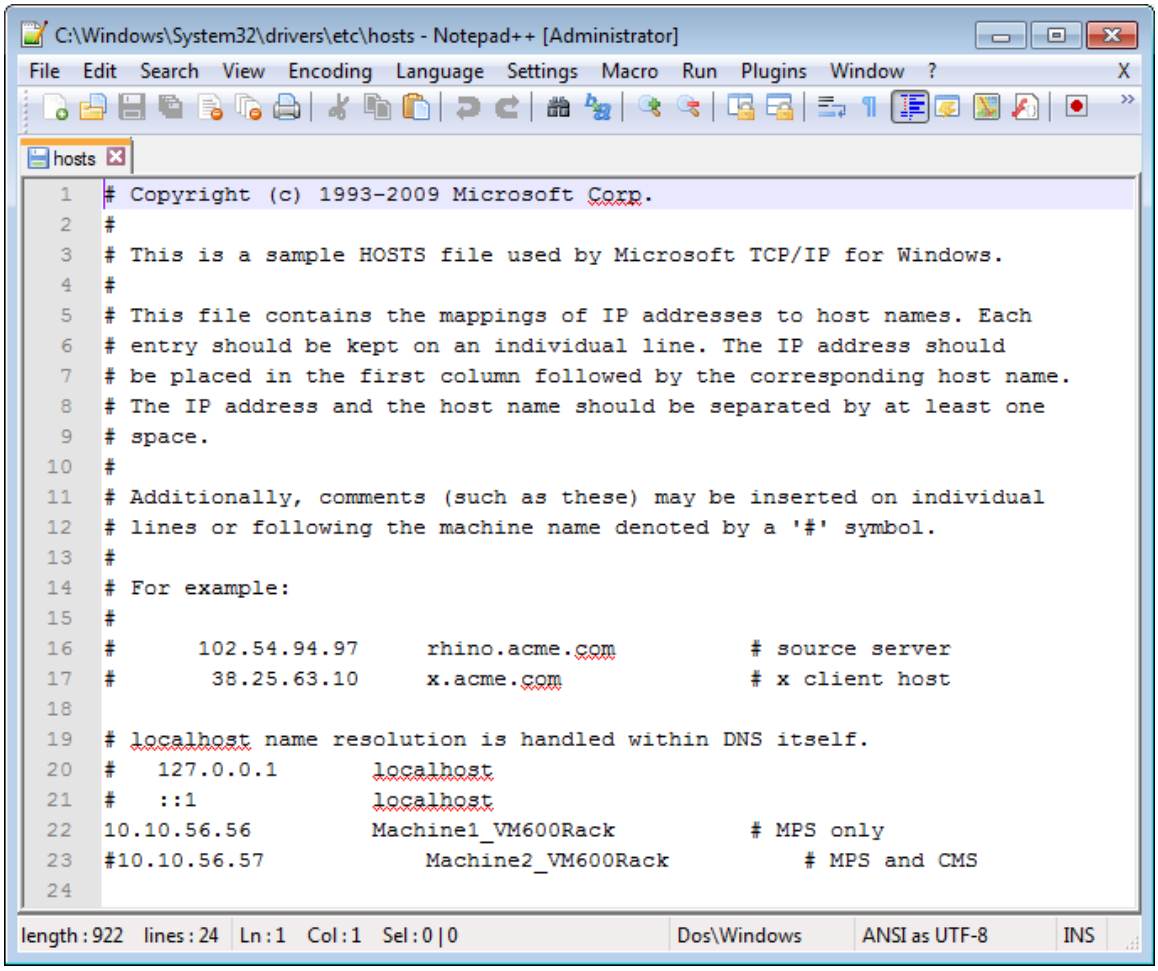



**Figure 4-4:** Exemple de fichier `hosts` Windows

Le fichier `hosts` représenté sur la Figure 4-4 inclut les noms d'hôte de deux racks VM600, par exemple :

- `Machine1_VM600Rack` avec l'adresse IP `10.10.56.56`
- `#Machine2_VM600Rack` avec l'adresse IP `10.10.56.57` (faisant actuellement l'objet d'un commentaire).

**NOTE :** Le caractère # est utilisé pour les commentaires de ligne et permet de « mettre en commentaire » de manière temporaire une ligne jusqu'à ce qu'elle soit utile dans le futur.

**3-** Modifiez le fichier `hosts` au besoin, en ajoutant une nouvelle ligne après les entrées `localhost`, pour chaque rack VM600 en utilisant son adresse IP (carte CPUM) et un nom d'hôte approprié.

**NOTE :** L'adresse IP du rack VM600 (carte CPUM) n'est pas réellement définie dans le fichier « hosts » de l'ordinateur (local), mais dans un fichier différent sur la mémoire de la carte CPUM (CompactFlash ou DiskOnChip).
Voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM.

**4-** Après sa modification, enregistrez le fichier `hosts` sous forme de fichier texte sans extension de nom de fichier (autrement dit, n'utilisez pas accidentellement l'extension `*.txt` ou toute autre extension de nom de fichier).

**NOTE :** L'ordinateur doit être redémarré pour que toutes les modifications apportées au fichier `hosts` entrent en vigueur (c'est-à-dire pour qu'elles soient enregistrées par Windows).

Pour un exemple d'utilisation des noms d'hôtes, voir la section 4.4 Utilisation du logiciel VM600 avec un rack en réseau.

### 4.2.4 Test des communications

Ping est un utilitaire d'administration de réseau utilisé pour tester l'accessibilité d'un hôte sur un réseau IP (protocole Internet), c'est-à-dire qu'il vérifie qu'un périphérique est connecté au même sous-réseau qu'un ordinateur. Il indique également le temps d'aller-retour pour les messages envoyés depuis l'hôte d'origine vers un périphérique de destination.

Le programme ping peut être exécuté à partir d'une invite de commande Windows. Pour tester la communication entre un ordinateur et une carte CPUM :

**1-** Démarrez une fenêtre d'invite de commande de l'une des manières suivantes :

Cliquez sur **Start > Programs > Accessories > Command Prompt**.

Ou cliquez sur **Start > Run**, puis tapez `cmd` dans la zone de texte Open et cliquez ensuite sur **OK**.

**2-** À l'invite de commande, tapez `ping xxx.xxx.xxx.xxx`, où `xxx.xxx.xxx.xxx` est remplacé par l'adresse IP de l'appareil en cours de vérification. Par exemple, un rack VM600 (carte CPUM) à l'adresse 10.10.56.56 et sur le même réseau que l'ordinateur.

**3-** Comme indiqué sur la Figure 4-4, si la configuration est correcte, l'appareil répond, ce qui indique qu'il se trouve dans le même sous-réseau que votre ordinateur et qu'il est prêt à être utilisé.

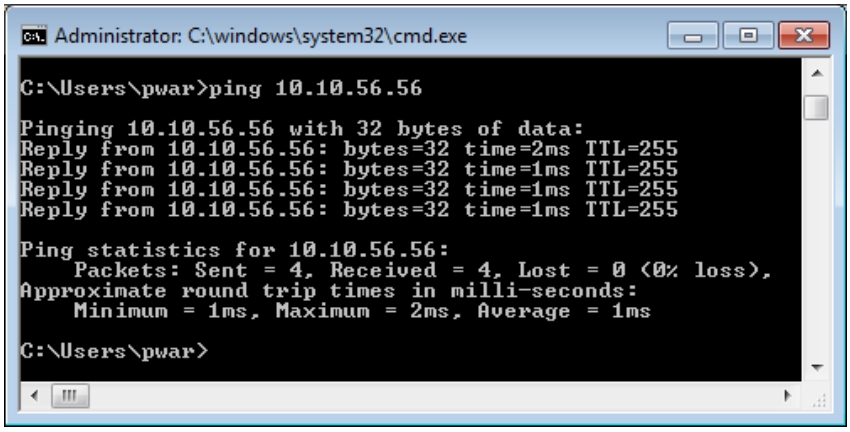



**Figure 4-5:** Ping d'un rack VM600 (carte CPUM)

**NOTE :** Pour pouvoir communiquer avec un appareil, l'ordinateur et l'appareil doivent être connectés au même réseau et posséder des adresses IP qui appartiennent au même sous-réseau.

Le programme ping est également inclus dans le microprogramme de la carte CPUM, de sorte que la commande `ping` peut être exécutée dans une session d'émulation de terminal afin de tester la communication entre une carte CPUM et un autre périphérique réseau, si nécessaire.

La touche de raccourci CTRL + C peut être utilisée pour arrêter une commande ping dans une session d'émulation de terminal.

## 4.3 Redondance Ethernet

Les cartes CPUM sont disponibles dans différentes versions, y compris standard, Ethernet redondante et série redondante.

**NOTE :** La redondance Ethernet est uniquement possible avec les versions Ethernet redondantes de la carte CPUM, c'est-à-dire avec des cartes équipées de deux interfaces Ethernet.
Reportez-vous à la fiche de données de la paire de cartes CPUM et IOCN pour plus d'informations.

Pour une configuration Ethernet redondante, deux adresses IP différentes sont attribuées à une seule paire de cartes CPUM / IOCN. Ces adresses doivent être choisies avec soin et doivent respecter les deux principes suivants :

**1-** Les deux adresses IP de la carte CPUM doivent être définies dans des sous-réseaux différents. Pour plus d'informations sur les sous-réseaux, voir la section 1.4.5.3 Sous-réseaux et masques de sous-réseaux.

**2-** Un ordinateur hôte qui communique avec l'une des adresses CPUM doit avoir son adresse IP dans le même sous-réseau que l'adresse CPUM correspondante.

Comme indiqué dans la section 1.4.5 Adressage IP, une adresse IP se divise en une partie sous-réseau (réseau) et une partie hôte. La plage de sous-réseau est définie par le masque de réseau.

Le Tableau 4-3 présente certaines valeurs de masque de réseau type et les adresses IP correspondantes communément utilisées.

**Tableau 4-3 :** Exemples de masques de réseau et adresses IP correspondantes

| Masque de réseau | | Adresse IP |
|---|---|---|
| **Notation hexadécimale** | **Notation décimale à point** | |
| 0xFF000000 | 255.0.0.0 | nn.hh.hh.hh |
| 0xFFFF0000 | 255.255.0.0 | nn.nn.hh.hh |
| 0xFFFFFF00 | 255.255.255.0 | nn.nn.nn.hh |
| 0xFFFFFF80 | 255.255.255.128 | nn.nn.nn.0-127/128-255<br>(n = sous-réseau, h = hôte) |

Le Tableau 4-4 présente certaines valeurs de masque de réseau et d'adresses IP valides pour une version Ethernet redondante de la carte CPUM et des deux ordinateurs hôtes.

**Tableau 4-4 :** Exemples de masques de réseau et adresses IP valides correspondantes pour une version Ethernet redondante de la carte CPUM et d'un ordinateur hôte

| | Masque de réseau | Adresses IP | |
|---|---|---|---|
| | | **Carte CPUM (et interface réseau)** | **Ordinateur hôte** |
| Exemple 1 | 0xFFFFFF80<br>(255.255.255.128) | 10.10.56.56 (principale)<br>10.10.56.156 (secondaire) | 10.10.56.1<br>10.10.56.129 |
| Exemple 2 | 0xFFFFFF00<br>(255.255.255.0) | 10.10.56.56 (principale)<br>10.10.57.56 (secondaire) | 10.10.56.1<br>10.10.57.1 |

### 4.3.1 Modification des adresses IP du rack

Comme indiqué dans le Tableau 4-3 et le Tableau 4-4, considérons un sous-réseau nn.nn.nn.hh avec le masque de réseau et les adresses IP suivants :

- Adresse IP de l'interface réseau principale (ETH1) : 10.10.56.56
- Adresse IP de l'interface réseau secondaire (ETH2) : 10.10.56.156
- Masque de réseau : 0xFFFFFF80

#### 4.3.1.1 Cartes CPUMexécutant la version 075 ou ultérieure du microprogramme

Pour une carte CPUM qui exécute la version 075 ou ultérieure du microprogramme, les informations d'interface réseau (adressage IP) sont définies dans le fichier `net.cfg` (voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM).

Pour modifier les adresses IP de la carte CPUM, le fichier `net.cfg` doit être modifié, au moyen du logiciel CPUM Configurator, par exemple (voir la section 6 CPUM Configurator et la section 6.8 Modification de l'adresse IP d'un rack VM600 (carte CPUM)) ou d'une

session d'émulation de terminal (voir la section 7 Émulation de terminal et la section 7.10 Modification de l'adresse IP d'un rack VM600 (carte CPUM)).

Comme indiqué sur la Figure 4-6 (partie supérieure), les paramètres affichés dans la boîte de dialogue **CPUM TCP-IP Configuration** de CPUM Configurator correspondent aux interfaces Ethernet de la carte CPUM comme suit :

- `Primary Ethernet Controller` pour l'interface réseau principale (ETH1).
- `Secondary Ethernet Controller` pour l'interface réseau secondaire (ETH2).

La modification des valeurs dans CPUM Configurator modifie les valeurs du fichier `net.cfg` sous-jacent.

Comme indiqué sur la Figure 4-6 (partie inférieure), les paramètres affichées dans le fichier `net.cfg` correspondent aux interfaces Ethernet de la carte CPUM comme suit :

- `#main network connection` et `net0_`… pour l'interface réseau principale (ETH1).
- `#redundant network connection` et `net1_`… pour l'interface réseau secondaire (ETH2).

Pour plus d'informations sur le fichier `net.cfg`, voir la section 3.4.2 À propos du fichier net.cfg.

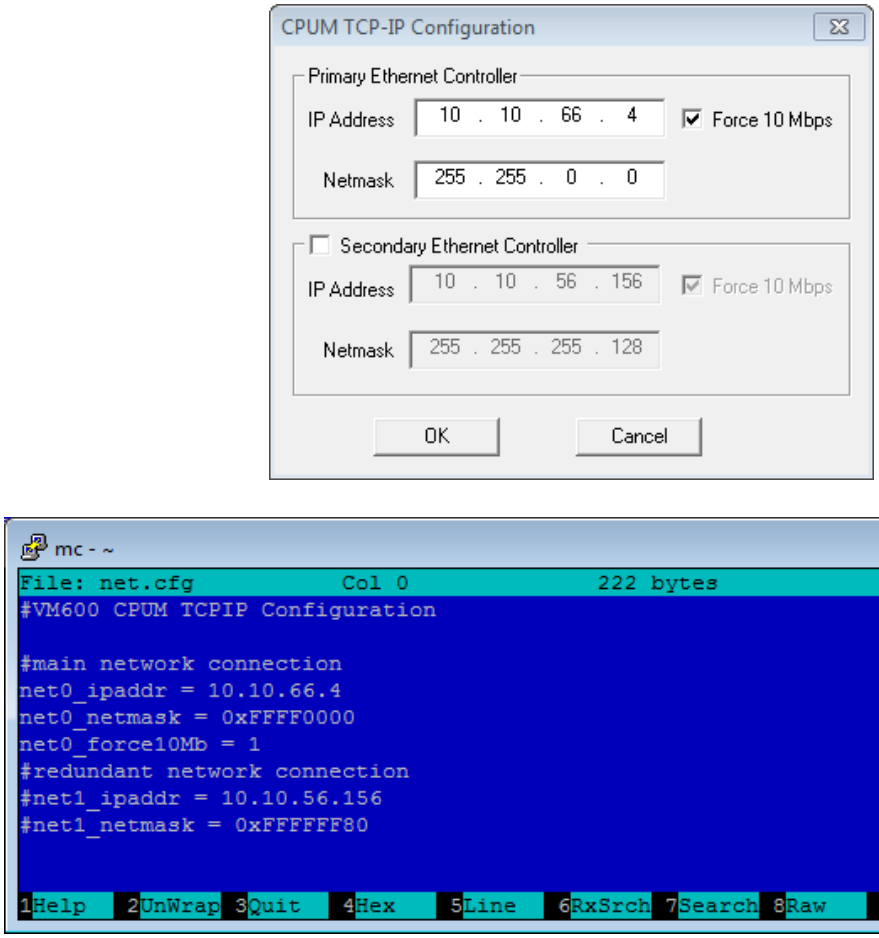



**Figure 4-6:** Informations d'interface réseau (adressage IP) provenant du fichier `net.cfg` d'une carte CPUM affichée dans CPUM Configurator (partie supérieure)

Terminez ensuite la procédure en redémarrant le rack : éteignez le rack VM600, patientez 2 à 3 secondes, puis rallumez le rack VM600. Si vous n'êtes pas en mesure de mettre physiquement le rack hors tension, vous pouvez utiliser la commande d'arrêt à la place.

Après la configuration correcte de la carte CPUM, les adresses IP des deux ordinateurs connectés aux deux ports Ethernet doivent être définies comme indiqué dans la section 4.2.2 Configuration de la carte réseau d'un ordinateur.

**NOTE :** L'adresse IP de chaque ordinateur doit être choisie dans le même sous-réseau que l'adresse IP de la carte CPUM correspondante.

L'adresse IP du premier ordinateur correspondant à l'adresse CPUM principale (10.10.56.56) peut être définie sur 10.10.56.55 et la deuxième sur 10.10.57.55, par exemple.

#### 4.3.1.2 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme

Pour une carte CPUM exécutant la version 074 ou antérieure du microprogramme, les informations d'interface réseau (adressage IP) sont définies dans le fichier `hosts` et le masque de réseau est défini dans le fichier `tcpip.1` (voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM).

Pour modifier les adresses IP de la carte CPUM, les fichiers `hosts` et `tcpip.1` doivent être modifiés, au moyen d'une session d'émulation de terminal, par exemple (voir la section 7 Émulation de terminal).

**NOTE :** Reportez-vous au *Manuel de mise en réseau du VMVM600, édition 6 (ou antérieure),* pour plus d'informations.

## 4.4 Utilisation du logiciel VM600 avec un rack en réseau

### 4.4.1 Logiciel VM600 MPS

**1-** Démarrez le logiciel VM600 MPS.

**NOTE :** Reportez-vous au *Manuel du logiciel des systèmes de protection des machines d'un logiciel de configuration VM600 MPSx* pour plus d'informations.

**2-** Dans l'arborescence (à gauche), sélectionnez un rack VM600 comme indiqué sur la Figure 4-7.

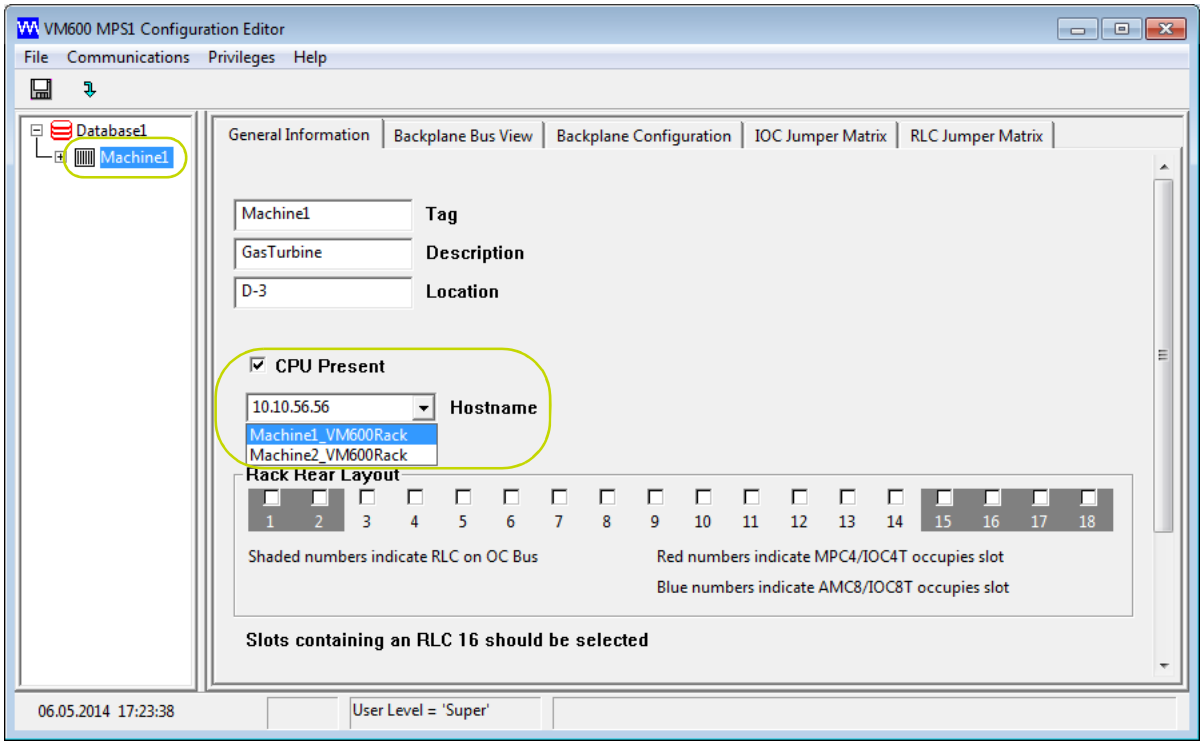



**Figure 4-7:** Utilisation de noms d'hôtes dans le logiciel VM600 MPS

**3-** Dans la fenêtre principale (au centre), sur l'onglet **General Information** :

Cochez la case **CPU Present**.

Pour **Hostname**, utilisez la liste déroulante pour sélectionner le nom d'hôte correspondant au rack VM600, qui a été défini dans le fichier hosts de Windows (voir la section 4.2.3 Modification du fichier hosts d'un ordinateur).

Vous pouvez aussi utiliser l'adresse IP du rack VM600 (10.10.56.56, par exemple) en la saisissant directement dans le champ **Hostname**.

**4-** La communication entre le logiciel VM600 MPS et le rack VM600 sur Ethernet doit désormais être possible.

### 4.4.2 Logiciel VM600 CMS

**1-** Démarrez le logiciel VM600 CMS.

**NOTE :** Reportez-vous au *Manuel du logiciel des systèmes de protection des machines d'un logiciel de VM600 CMS* pour plus d'informations.

**2-** Dans l'arborescence **Architectural View** (à gauche), sélectionnez un rack VM600 comme indiqué sur la Figure 4-8.

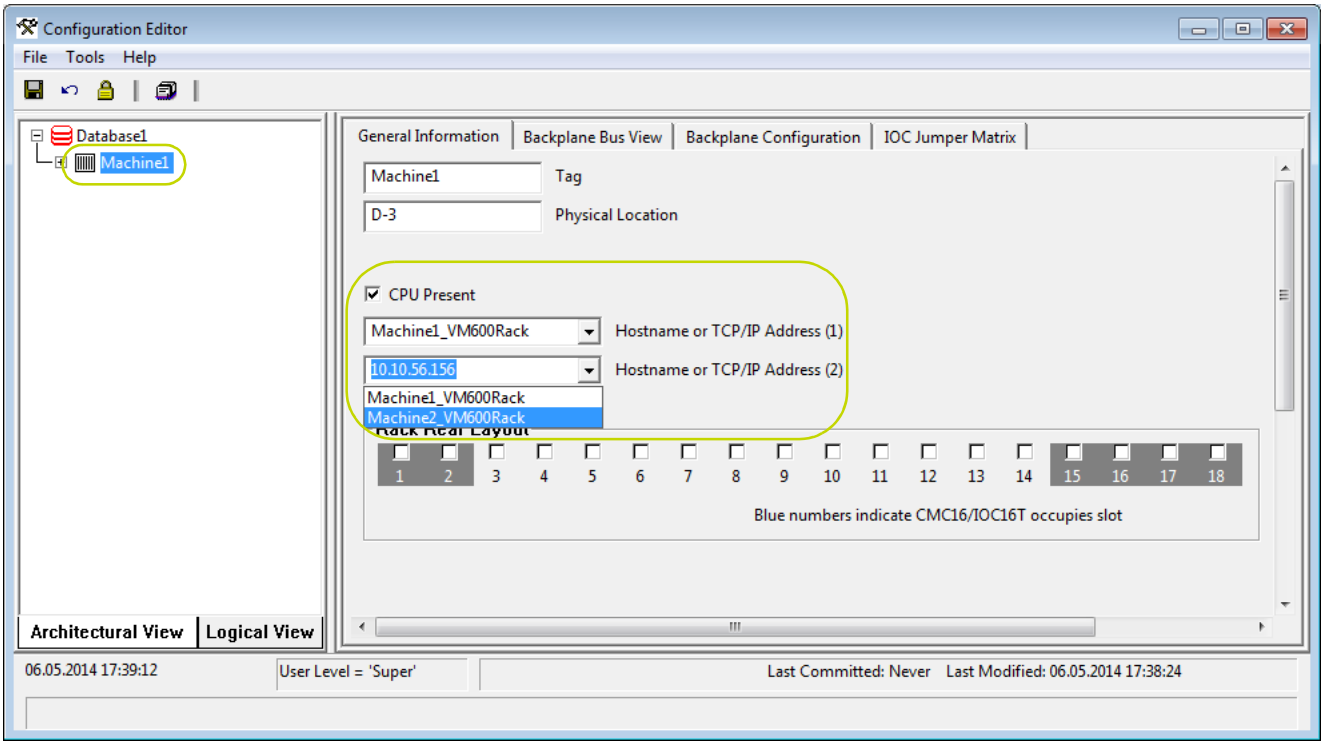



**Figure 4-8:** Utilisation de noms d'hôtes dans le logiciel VM600 CMS

**3-** Dans la fenêtre principale (au centre), sur l'onglet **General Information** :

Cochez la case **CPU Present**.

Pour les champs **Hostname ou TCP**/IP address, utilisez la liste déroulante pour sélectionner le nom d'hôte correspondant au rack VM600, qui a été défini dans le fichier hosts de Windows (voir la section 4.2.3 Modification du fichier hosts d'un ordinateur).

Vous pouvez aussi utiliser l'adresse IP du rack VM600 (10.10.56.56, par exemple) en la saisissant directement dans le champ **Hostname ou TCP**/IP address.

**4-** La communication entre le logiciel VM600 CMS et le rack VM600 sur Ethernet doit désormais être possible.

# 5 COMMUNICATIONS ETHERNET AVEC UN RACK VM600

Il existe de nombreuses façons de communiquer avec des racks VM600 et de les configurer.

**NOTE :** Reportez-vous au *Manuel du matériel du système de protection des machines (MPS) du VM600*pour plus d'informations sur les racks en réseau et les racks autonomes.

Historiquement, les racks VM600 étaient gérés à l'aide d'un programme d'émulation de terminal (par exemple, PuTTY ou HyperTerminal) exécuté sur un ordinateur pour établir des communications avec le rack VM600, puis en exécutant l'éditeur de texte vi fourni par le microprogramme de la carte CPUM pour modifier les fichiers de configuration si nécessaire.

**NOTE :** Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations sur l'utilisation de HyperTerminal et de l'éditeur de texte vi pour gérer des racks VM600 en réseau.

Cependant, la gestion de racks VM600 en réseau est nettement simplifiée en faisant appel à des communications Ethernet telles que :

- Le logiciel CPUM Configurator exécuté sur un ordinateur (voir la section 6 CPUM Configurator).
- Une session d'émulation de terminal sur un ordinateur exécutant Midnight Commander sur la carte CPUM (voir la section 7 Émulation de terminal).

Cependant, pour que les communications Ethernet puissent être utilisées, l'adresse IP de la carte CPUM dans le rack VM600 en réseau doit être connue. Cette information peut être obtenue directement de la carte CPUM au moyen d'une communication série (voir la section 3 Communications série avec un rack VM600).

**NOTE :** Pour utiliser le logiciel CPUM Configurator ou une connexion Telnet à un rack VM600, l'adresse IP de la carte CPUM doit être connue.
Si l'adresse IP de la carte CPUM n'est pas connue, une connexion série (RS-232) à la carte peut être utilisée pour découvrir l'adresse IP de la carte.
Voir la section 3.4 Découverte de l'adresse IP d'une carte CPUM.

En outre, l'ordinateur utilisé pour communiquer avec le rack VM600 (carte CPUM) doit être correctement configuré pour permettre les communications Ethernet avec la carte CPUM. Ceci signifie que l'adaptateur réseau (carte d'interface réseau) de l'ordinateur doit être configuré pour être compatible avec l'adresse IP et le masque de sous-réseau utilisés par la carte CPUM, c'est-à-dire que les deux périphériques doivent se trouver dans le même sous-réseau. Voir la section 4 Configuration d'une connexion Ethernet pour plus d'informations.

## 5.1 Établissement de communications Ethernet avec une carte CPUM

La procédure exacte à suivre dépend du système d'exploitation et des outils logiciels utilisés. La description qui suit utilise l'émulateur de terminal PuTTY sous Windows® 7. Si vous utilisez un système d'exploitation différent ou un émulateur de terminal prenant en charge le protocole Telnet, vous pouvez quand même suivre cette description à titre indicatif.

En cas de questions ou de problèmes, consultez l'aide de Windows ou contactez votre administrateur système. Si aucune solution ne peut être trouvée, contactez votre représentant Meggitt le plus proche.

**NOTE :** Si votre carte CPUM ne gère pas les communications Ethernet, une connexion PPP peut être utilisée pour communiquer avec le rack pour les cartes exécutant la version 072 ou antérieure du microprogramme. Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations sur le protocole PPP.

Pour une connexion directe entre un rack VM600 et un ordinateur, utilisez un câble Ethernet croisé entre le connecteur 8P8C (**NET**) sur le panneau avant de la carte CPUM ou le connecteur 8P8C supérieur (**1**) sur le panneau avant de la carte IOCN et le port Ethernet de l'ordinateur (pas un câble standard).

**NOTE :** L'interface réseau principale (ETH1) d'une carte CPUM peut être acheminée via le connecteur 8P8C (**NET**) qui se trouve sur le panneau avant de la carte CPUM ou via le connecteur 8P8C supérieur (**1**) qui se trouve sur le panneau avant de la carte IOCN associée (en option). Le choix est fait au moyen des cavaliers J42, J40, J52 et J53 de la carte CPUM. Lorsque tous ces cavaliers sont retirés, par exemple, l'interface réseau principale est acheminée via la carte CPUM.
L'interface réseau secondaire (ETH2) d'une carte CPUM est acheminée via le connecteur 8P8C inférieur (**2**) qui se trouve sur le panneau avant de la carte IOCN associée (en option).
Reportez-vous au *Manuel du matériel du système de protection des machines (MPS) du VM600* pour plus d'informations.

Pour une connexion indirect entre une carte CPUM et un ordinateur via un réseau (par l'intermédiaire de commutateurs réseau), utilisez des câbles Ethernet standard (pas de câbles croisés).

Voir la section 4 Configuration d'une connexion Ethernet.

## 5.2 Utilisation d'une connexion Ethernet pour vérifier l'adresse IP d'une carte CPUM

Si l'adresse IP d'une carte CPUM n'est pas connue, elle peut malgré tout être découverte par l'intermédiaire d'une connexion série avec le connecteur RS-232 sur le panneau avant de la carte CPUM et d'un logiciel d'émulation de terminal (voir la section 3.4 Découverte de l'adresse IP d'une carte CPUM).

**NOTE :** La structure des répertoires et les fichiers de configuration dépendent de la version du microprogramme exécutée sur une carte CPUM (voir la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM). Pour une carte CPUM qui exécute la version 075 ou ultérieure du microprogramme, les informations d'adresse IP sont définies dans le fichier `net.cfg`.

### 5.2.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme

**1-** Connectez un rack VM600 (carte CPUM) et un ordinateur au moyen du câble Ethernet approprié.

**2-** Démarrez PuTTY, sélectionnez une connexion **Telnet** et entrez l'adresse IP (**Nom d'hôte (ou adresse IP)**) de la carte CPUM, comme indiqué sur la Figure 5-1. (Telnet utilise toujours le numéro de port TCP 23 (**Port**).)

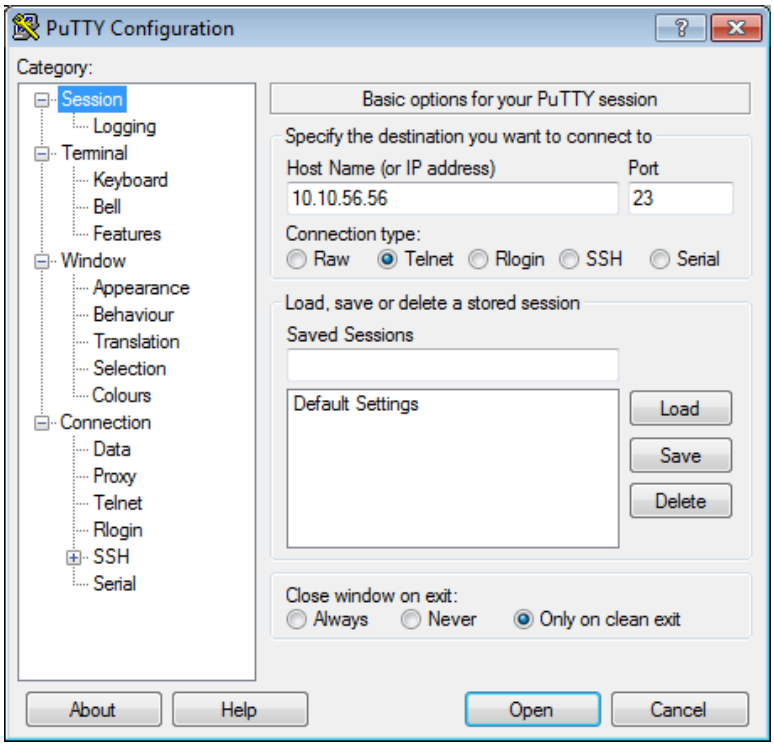



**Figure 5-1:** Utilisation du logiciel d'émulation de terminal PuTTY – sélection et configuration d'une connexion Telnet

**3-** Acceptez les options de session Telnet par défaut de PuTTY, comme indiqué sur la Figure 5-2.

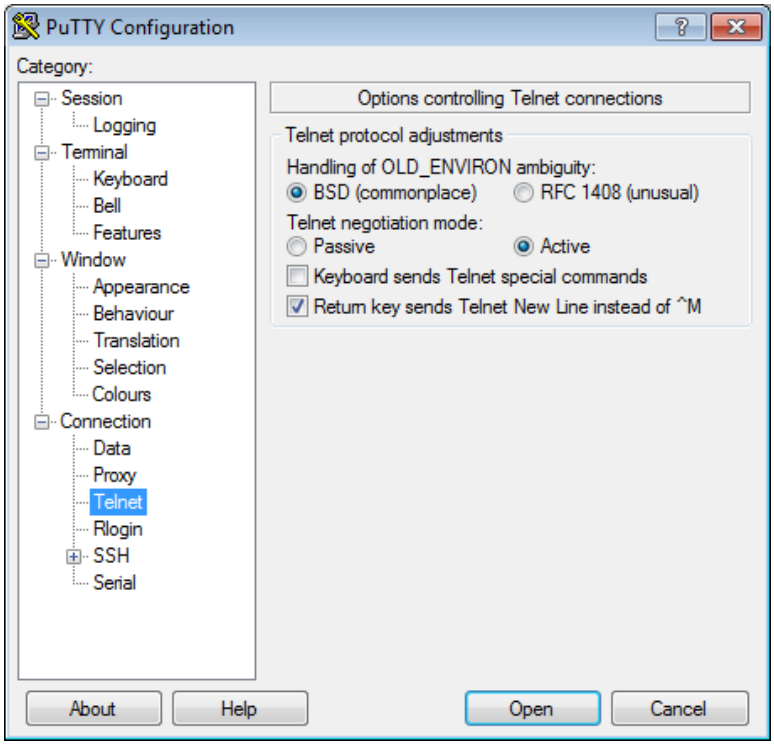



**Figure 5-2:** Utilisation du logiciel d'émulation de terminal PuTTY – options Telnet par défaut

**4-** Cliquez sur **Open** pour démarrer la session PuTTY, puis appuyez sur la touche ENTRÉE.

**5-** Lorsque vous êtes invité à vous connecter, tapez **user** (identifiant de connexion par défaut), puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
login : user
```

Lorsque vous êtes invité à saisir un mot de passe, tapez **config** (mot de passe par défaut), puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
password : config
```

**6-** À l'invite de commande ( `/usr/user #` ), tapez **ls**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # ls
```

La commande list dresse la liste des fichiers du répertoire de travail en cours ( `/usr/user` ), y compris le fichier `version.txt` qui contient les informations de version du microprogramme pour une carte CPUM.

**7-** À l'invite de commande ( `/usr/user #` ), tapez **more net.cfg**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # more net.cfg
```

La commande `more` affiche le contenu du fichier texte spécifié écran par écran. Si nécessaire, appuyez sur ENTRÉE ou ESPACE pour afficher les *n* lignes de texte suivantes.

Vous pouvez également utiliser la commande `cat` pour afficher le contenu d'un fichier sans pause.

**8-** Dans le fichier `net.cfg`, les adresses IP des interfaces réseau de la carte CPUM sont affichées sous `#main network connection` sous la forme `net0_ipaddr` (pour l'interface réseau principale (ETH1)) et sous `#redundant network connection` sous la forme `net1_ipaddr` (pour l'interface réseau secondaire (ETH2)).

Voir la section 3.4.2 À propos du fichier net.cfg pour plus d'informations sur le fichier `net.cfg`.

### 5.2.2 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme

Pour une carte CPUM exécutant la version 074 ou antérieure du microprogramme, les adresses IP sont définies dans le fichier `hosts`. La procédure décrite dans la section 5.2.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme peut être utilisée, mais en accédant au fichier `hosts` (plutôt qu'au fichier `net.cfg`).

Par exemple, l'adresse IP de l'interface réseau de la carte CPUM est affichée sous `#Primary Ethernet Port` sous la forme `node1`.

**NOTE :** Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations.

## 5.3 Utilisation d'une connexion Ethernet pour identifier la version du microprogramme d'une carte CPUM

Il est important de connaître la version du microprogramme exécuté sur une carte CPUM dans la mesure où les fonctions de mise en réseau, la compatibilité logicielle et les fichiers de configuration et la structure peuvent différer, en fonction de la version du microprogramme.

Pour toute carte CPUM, la version du microprogramme figure dans le fichier `version.txt`.

La version du microprogramme exécuté sur une carte CPUM peut être identifiée à l'aide d'une connexion série ou d'une connexion Ethernet avec la carte CPUM et d'un logiciel d'émulation de terminal.

Pour plus d'informations sur l'établissement d'une connexion série avec une carte CPUM, voir la section 3.2 Établissement de communications série avec une carte CPUM.

Pour plus d'informations sur l'établissement d'une Ethernet série avec une carte CPUM, voir ci-dessous.

**1-** Connectez un rack VM600 (carte CPUM) et un ordinateur au moyen du câble Ethernet approprié.

**2-** Démarrez PuTTY, sélectionnez une connexion **Telnet** et entrez l'adresse IP (**Nom d'hôte (ou adresse IP)**) de la carte CPUM, comme indiqué sur la Figure 5-3. (Telnet utilise toujours le numéro de port TCP 23 (**Port**).)

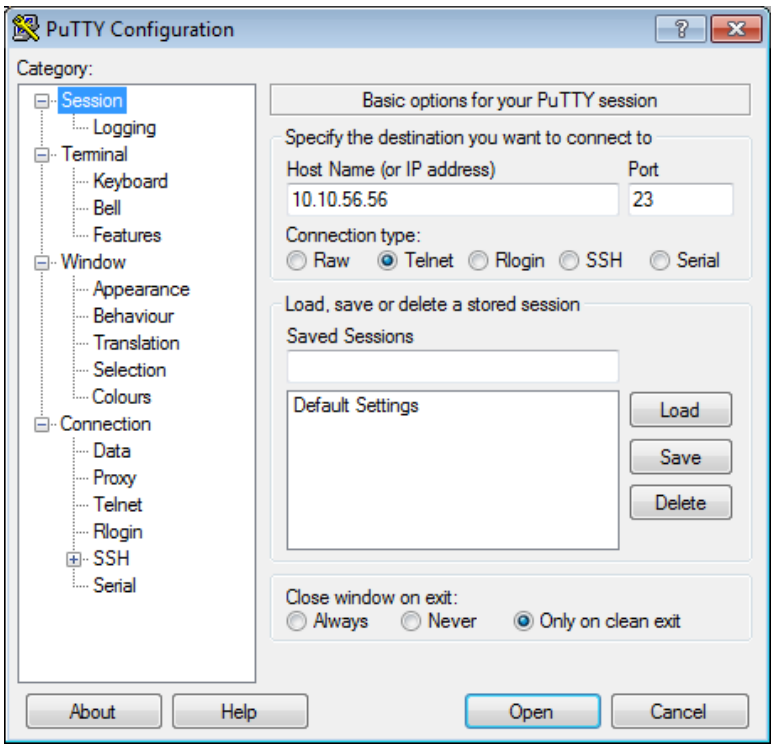



**Figure 5-3:** Utilisation du logiciel d'émulation de terminal PuTTY – sélection et configuration d'une connexion Telnet

**3-** Acceptez les options de session Telnet par défaut de PuTTY, comme indiqué sur la Figure 5-4.

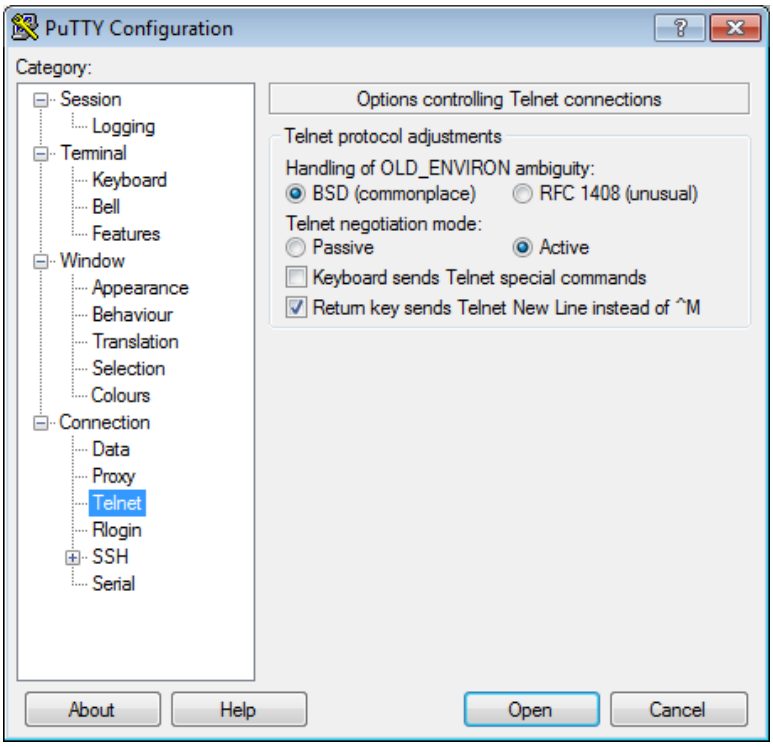



**Figure 5-4:** Utilisation du logiciel d'émulation de terminal PuTTY – options Telnet par défaut

**4-** Cliquez sur **Open** pour démarrer la session PuTTY, puis appuyez sur la touche ENTRÉE.

**5-** Lorsque vous êtes invité à vous connecter, tapez **user** (identifiant de connexion par défaut), puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
login : user
```

Lorsque vous êtes invité à saisir un mot de passe, tapez **config** (mot de passe par défaut), puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
password : config
```

**6-** À l'invite de commande ( `/usr/user #` ), tapez **ls**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # ls
```

La commande list dresse la liste des fichiers du répertoire de travail en cours ( `/usr/user` ), y compris le fichier `version.txt` qui contient les informations de version du microprogramme pour une carte CPUM.

**7-** À l'invite de commande ( `/usr/user #` ), tapez **more version.txt**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # more version.txt
```

La commande more affiche le contenu du fichier texte spécifié écran par écran. Si nécessaire, appuyez sur ENTRÉE ou ESPACE pour afficher les *n* lignes de texte suivantes.

Voir la section 3.3.1 À propos du fichier version.txt pour plus d'informations sur le fichier `version.txt`.

## 5.4 Modification des fichiers de configuration

Voir la section 2.5 Utilisation des fichiers de configuration.

## 5.5 Passerelles

De façon générale, une passerelle est nécessaire pour permettre la communication entre deux réseaux ou sous-réseaux différents (voir la section 1.4.2 Connectivité au sein des réseaux).

Dans l'exemple de la Figure 5-5, le système 1 peut communiquer directement avec le système 2, car ils possèdent tous deux une adresse IP commençant par 10.10. Cependant, le système 1 ne peut pas communiquer directement avec le système 3 parce que les adresses IP sont trop différentes. Une passerelle est donc nécessaire.

Le système 1 peut communiquer avec la passerelle car ils possèdent tous deux une adresse IP commençant par 10.10. De même, le système 3 peut communiquer avec la passerelle car ils possèdent tous deux une adresse IP commençant par 194.11.227.

**NOTE :** Une passerelle doit appartenir au même sous-réseau que le réseau pour lequel elle fait office d'intermédiaire, c'est-à-dire que le même masque de sous-réseau s'applique.

Notez que si le masque de sous-réseau du sous-réseau « A » passait de 255.255.0.0 à 255.255.255.0, la communication ne serait plus possible entre le système 1 et le système 2 ou entre le système 1 et la passerelle.

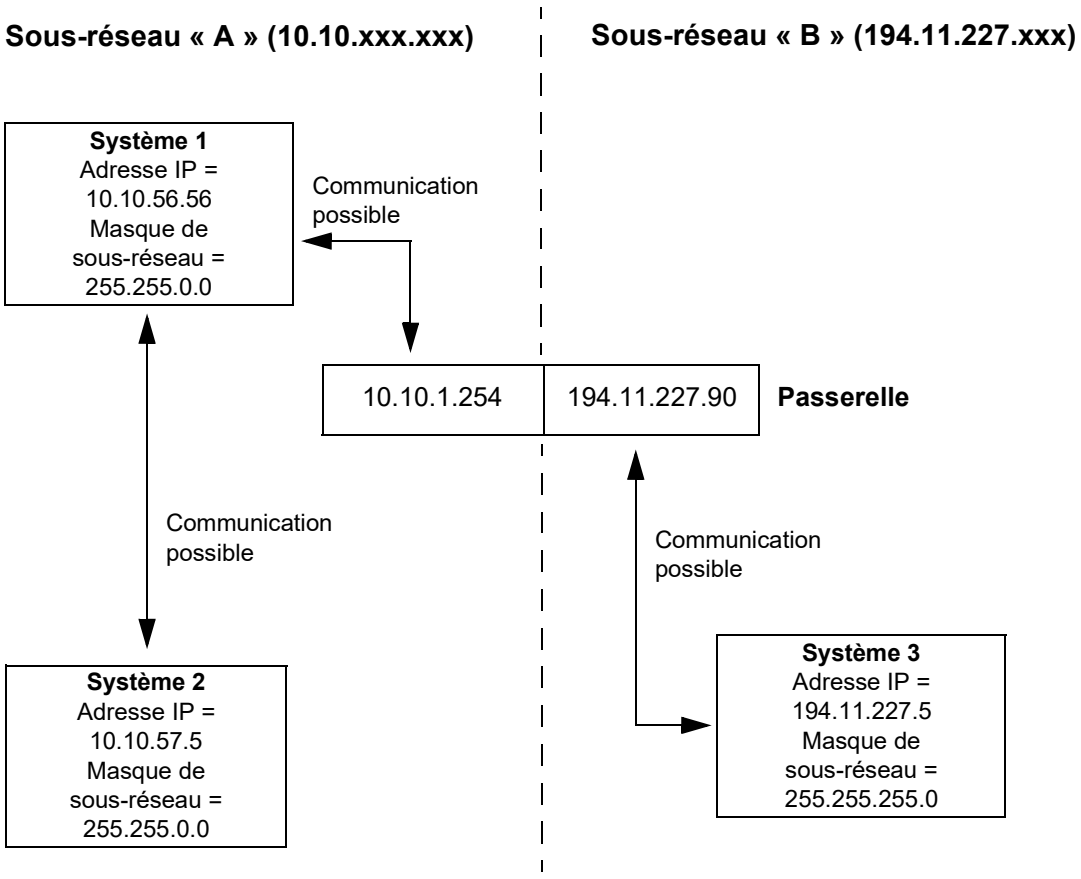



**Figure 5-5:** Communication entre deux sous-réseaux au moyen d'une passerelle

Pour un rack VM600, la communication Ethernet entre l'ordinateur hôte et le rack VM600 peut seulement avoir lieu si les adresses IP de l'ordinateur hôte et du rack VM600 (carte CPUM) appartiennent au même sous-réseau.

Pour contourner ce problème potentiel, une passerelle par défaut peut être définie pour une carte CPUM.

**NOTE :** Pour un bon fonctionnement, les adresses IP de la passerelle et de la carte CPUM doivent appartenir au même sous-réseau. Si ce n'est pas le cas, la passerelle n'est pas accessible à la carte CPUM.

De plus, l'ordinateur hôte et la passerelle doivent également posséder des adresses IP mutuellement visibles.

## 5.6 Ajout d'une passerelle par défaut à une carte CPUM

### 5.6.1 Cartes CPUM exécutant la version 075 ou ultérieure du microprogramme

Pour une carte CPUM exécutant la version 075 ou ultérieure du microprogramme, la passerelle par défaut d'une carte CPUM est définie et ajoutée au moyen du fichier `sysinit`.

### 5.6.2 À propos du fichier sysinit

Le fichier `sysinit` est un fichier d'initialisation du système contenant des commandes permettant de configurer divers pilotes de périphériques et services pour la carte CPUM.

Pour ajouter une passerelle par défaut à une carte CPUM, une commande doit être ajoutée à la section `Configure TCP/IP` du fichier `sysinit`, comme indiqué dans la section 5.6.2.1 Exemple de fichier sysinit.

Le chemin d'accès au fichier `sysinit` est `/etc/system/sysinit`

#### 5.6.2.1 Exemple de fichier sysinit

Section `Configure TCP/IP` d'un exemple de fichier `sysinit` sans passerelle par défaut :

```
#--------------------------------
#
# Configure TCP/IP
startnet
```

Section `Configure TCP/IP` d'un exemple de fichier `sysinit` avec une passerelle par défaut :

```
#--------------------------------
#
# Configure TCP/IP
startnet
route add default 10.10.1.254
```

#### 5.6.2.2 Description

Pour ajouter une passerelle par défaut à une carte CPUM, la commande `route` est utilisée. Par exemple :

```
route add default 10.10.1.254
```

Où :

`route add default` est la commande permettant d'ajouter la passerelle par défaut.

`10.10.1.254` est l'adresse IP de la passerelle par défaut, en notation décimale à point.

Comme indiqué dans la section 2.5 Utilisation des fichiers de configuration, le moyen le plus simple de modifier le fichier `sysinit` est une session d'émulation de terminal sur un ordinateur exécutant Midnight Commander sur la carte CPUM (voir la section 7 Émulation de terminal).

---

**NOTE :** Comme `sysinit` est un fichier d'initialisation du système, il est lu et ses commandes sont exécutées au démarrage de la carte CPUM. La carte doit par conséquent être redémarrée pour que les modifications apportées à ce fichier entrent en vigueur.

---

### 5.6.3 Cartes CPUM exécutant la version 074 ou antérieure du microprogramme

Pour une carte CPUM exécutant la version 074 ou antérieure du microprogramme, la passerelle par défaut d'une carte CPUM est définie et ajoutée au moyen des fichiers `hosts` et `tcpip.1`.

---

**NOTE :** Reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure),* pour plus d'informations.

---

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# 6 CPUM CONFIGURATOR

Le logiciel CPUM Configurator simplifie la gestion des racks VM600 en réseau équipés d'une carte CPUM exécutant la version 075 ou ultérieure du microprogramme.

**NOTE :** CPUM Configurator est compatible avec les cartes CPUM exécutant la version 075 ou ultérieure du microprogramme.

Pour identifier la version du microprogramme d'une carte CPUM, voir la section 3.3 Identification de la version du microprogramme d'une carte CPUM.

Pour travailler avec les cartes CPUM exécutant la version 074 ou antérieure du microprogramme, voir la section 7 Émulation de terminal ou au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure)*.

## 6.1 À propos de CPUM Configurator

CPUM Configurator est un programme qui communique avec une carte CPUM dans un rack VM600 via une liaison Ethernet (TCP/IP).

Fondamentalement, il fournit une interface utilisateur graphique pour une session Telnet entre le logiciel CPUM Configurator (client Telnet) et une carte CPUM (serveur Telnet), et s'utilise principalement pour la configuration des cartes CPUM et la gestion des racks VM600 sur les liaisons Ethernet.

Le logiciel CPUM Configurator est un fichier programme Windows® individuel (`*.exe`).

### 6.1.1 Fichiers de configuration de la carte CPUM

Une carte CPUM utilise un certain nombre de fichiers de configuration différents pour configurer des fonctionnalités et des fonctions distinctes de la carte CPUM. Ce sont ces fichiers de configuration sous-jacents qui sont consultés, modifiés et/ou mis à jour au moyen de CPUM Configurator.

Les fichiers de configuration dépendent de l'application, et ils ne sont par conséquent pas tous utilisés par toutes les cartes CPUM. Par exemple, seul un système de surveillance assurant des fonctions de surveillance d'état au moyen de cartes de surveillance d'état XMx16 nécessite les fichiers de configuration `xmcsrv.cfg` et `xmcnn.xml`.

En règle générale, la configuration utilisateur complète d'une carte CPUM comprend les fichiers répertoriés dans le Tableau 6-1.

**Tableau 6-1 :** Fichiers de configuration des cartes CPUM

| **Carte CPUM exécutant la version 075 ou ultérieure du microprogramme** | |
|---|---|
| **Fichier de configuration** | **Description** |
| `launcher.cfg` | Processus de lancement (pour surveiller tous les autres processus) :<br>Informations de configuration pour le processus de lancement (qui surveille les autres processus) exécuté sur la carte CPUM. |
| `modbusDefault.cfg` | Communications Modbus et PROFINET :<br>Informations de configuration pour les communications Modbus et PROFINET, y compris les informations de mappage de registre pour l'interface Modbus et les informations sur les logements PROFINET pour l'interface PROFINET exécutée sur la carte CPUM. |
| `profinet.cfg` | Communications PROFINET :<br>Informations de configuration supplémentaires pour les communications PROFINET, y compris le nom du périphérique et la passerelle pour l'interface PROFINET exécutée sur la carte CPUM. |
| `net.cfg` | Adaptateurs réseau (adressage IP) :<br>Informations de configuration pour les interfaces Ethernet, y compris les informations d'adresse IP et de masque réseau (correspondant à la boîte de dialogue de configuration TCP-IP de la carte CPUM) |
| `ntp.conf` | Serveur NTP :<br>Informations de configuration du serveur NTP (Network Time Protocol) local exécuté la carte CPUM. |
| `uc.cfg` | Informations de résumé :<br>Informations de base sur la configuration utilisateur complète téléchargée à partir d'une carte CPUM (**Save UC from CPUM**). |
| `xmcnn.xml` | Carte XMx16 – individuelle :<br>Informations de configuration pour une carte XMx16 installée dans le logement *nn* du même rack VM600 que la carte CPUM qui communique avec le processus VibroSight Server exécuté sur la carte CPUM. |
| `xmcsrv.cfg` | Cartes XMx16 – résumé :<br>Informations de base sur la configuration des cartes XMx16 installées dans le même rack VM600 que la carte CPUM (correspondant à la boîte de dialogue de configuration de XMCSRV). |

Voir aussi la section 2 Fichiers de configuration et structure des répertoires de la carte CPUM.

## 6.2 Installation de CPUM Configurator

Le logiciel CPUM Configurator est inclus avec le logiciel VM600 MPSx version 2.7 ou ultérieure et est copié sur l'ordinateur dans le cadre de la procédure d'installation du logiciel VM600 MPSx.

CPUM Configurator est installé dans le même dossier que le logiciel VM600 MPSx sur le disque dur. Les dossiers par défaut sont :

- `C:\Program Files\VM600_MPS\Bin` – Il s'agit du dossier par défaut pour les fichiers programme du logiciel VM600 MPSx sur les versions 32 bits des systèmes d'exploitation Windows.
- `C:\Program Files (x86)\VM600_MPS\Bin` – Il s'agit du dossier par défaut pour les fichiers programme du logiciel VM600 MPSx sur les versions 64 bits des systèmes d'exploitation Windows.

D'autres dossiers peuvent être sélectionnés, le cas échéant.

Après son installation par le logiciel VM600 MPSx, CPUM Configurator apparaît dans le menu Démarrer de Windows mais aucun raccourci (icône) n'est ajouté au bureau Windows.

Le logiciel CPUM Configurator peut également être obtenu auprès du service clientèle de Meggitt (voir la section 12.1 Pour nous contacter), auquel cas le fichier programme de CPUM Configurator (`CPUMConfig.exe`) doit être copié dans un dossier sur le disque dur local de l'ordinateur.

## 6.3 Démarrage de CPUM Configurator

Pour démarrer CPUM Configurator :

**1-** Cliquez sur **Start > All Programs > VM600 MPS > CPUM Configurator**, ou accédez au fichier programme de CPUM Configurator (`CPUMConfig.exe`) sur le disque dur local de l'ordinateur et double-cliquez dessus.

**2-** Lorsque vous êtes invité à vous connecter, tapez `user` (identifiant de connexion par défaut) dans le champ **User name**, et `config` (mot de passe par défaut) dans le champ **Password**, puis cliquez sur **OK**.

Si vous cochez la case **Login automatically** avant de cliquer sur **OK**, CPUM Configurator mémorisera et utilisera automatiquement le nom d'utilisateur et le mot de passe lors du prochain démarrage de CPUM Configurator.

**3-** La fenêtre CPUM Configurator s'affiche alors, comme indiqué sur la Figure 6-1.

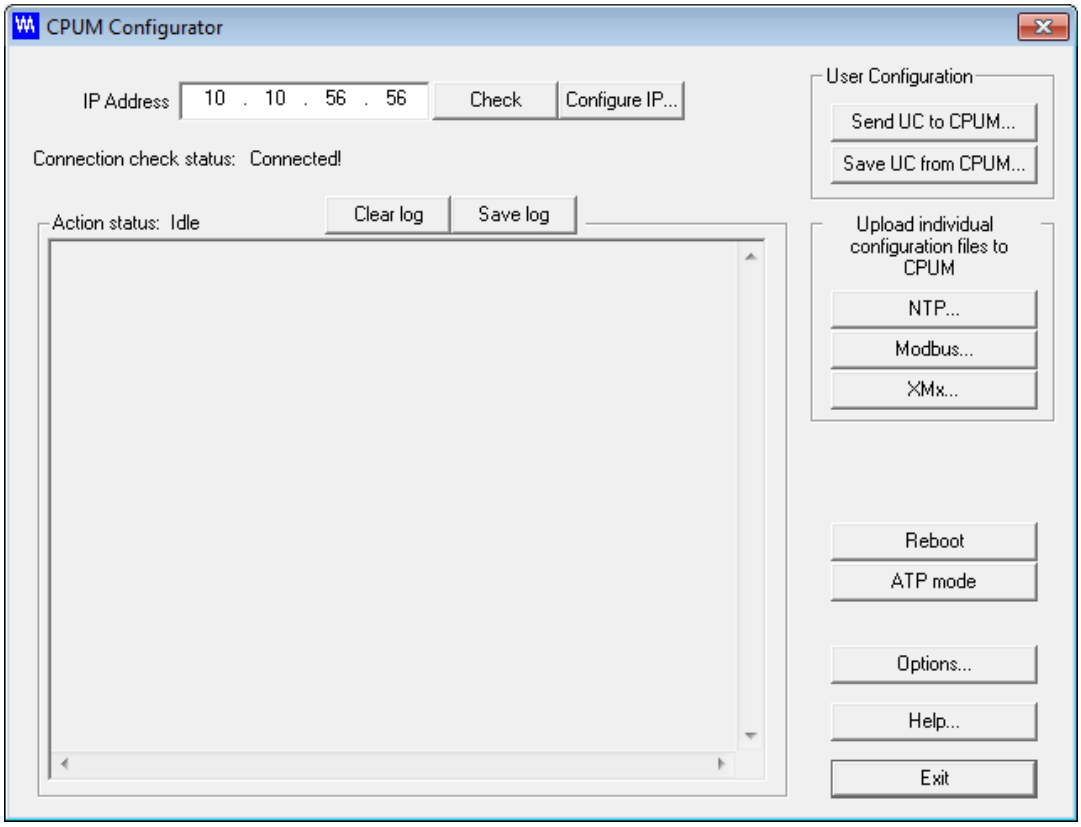



**Figure 6-1:** CPUM Configurator

## 6.4 Arrêt de CPUM Configurator

Pour fermer CPUM Configurator :

- Cliquez sur le bouton **Exit**.
- Ou cliquez sur la case de **fermeture** de Windows (en haut à droite).

Cette opération ferme la connexion à la carte CPUM et quitte CPUM Configurator.

## 6.5 Aide sur l'utilisation de CPUM Configurator

Pour obtenir de l'aide sur l'utilisation de CPUM Configurator :

- Cliquez sur le bouton **Help** ou appuyez sur la touche F1.

Vous devez fermer la fenêtre d'aide de CPUM Configurator affichée pour continuer à travailler avec CPUM Configurator.

## 6.6 Messages affichés lors de l'utilisation de CPUM Configurator

Lors de l'utilisation de CPUM Configurator, des messages sont communiqués au moyen de boîtes de dialogue individuelles et/ou dans la zone de texte Action status (voir la section 6.6.1 Zone de texte Action status).

### 6.6.1 Zone de texte Action status

L'état et les résultats des opérations effectuées par le logiciel CPUM Configurator et le microprogramme CPUM sont enregistrés dans la zone de texte **Action status**.

Certaines des opérations exécutées par une carte CPUM sont traitées en arrière-plan et peuvent demander un certain temps ; leur progression est de ce fait consignée ici.

De plus, d'autres messages à l'intention de l'utilisateur, tels que des instructions de redémarrage de la carte CPUM s'affichent ici.

### 6.6.2 Bouton Clear Log

Cliquez sur le bouton **Clear Log** pour effacer toutes les informations consignées dans la zone de texte Action status.

### 6.6.3 Bouton Save Log

Cliquez sur le bouton **Save Log** pour enregistrer les informations actuellement consignées dans la zone de texte Action dans un fichier texte (`*.txt`) sur l'ordinateur exécutant CPUM Configurator, pour référence future. (Autrement dit, toutes les informations consignées depuis le début de la session CPUM Configurator ou depuis la dernière opération Clear Log.)

La boîte de dialogue Save As qui apparaît permet de spécifier l'emplacement et le nom du fichier.

## 6.7 Établissement de communications avec une carte CPUM

Pour commencer à communiquer avec une carte CPUM, dans la fenêtre CPUM Configurator (voir la Figure 6-1) :

**1-** Dans la zone de texte **IP Address**, saisissez l'adresse IP de la carte CPUM en notation décimale à point. Exemple : `10.10.56.56`.

---

**NOTE :** Si l'adresse IP d'une carte CPUM n'est pas connue, elle peut être découverte au moyen d'une liaison série directe (RS-232) vers la carte CPUM. Voir la section 3.4 Découverte de l'adresse IP d'une carte CPUM.

---

**2-** Cliquez sur le bouton **Check** pour établir une connexion entre CPUM Configurator et la carte CPUM dont l'adresse est spécifiée dans la zone de texte IP Address.

L'état de la connexion entre CPUM Configurator et une carte CPUM s'affiche sous **Connection Check Status**.

**3-** Deux cas de figure sont alors possibles :

Le message `Connection Check Status: Connected!` s'affiche pour indiquer qu'une communication a été établie.

Le message `Connection Check Status: No response from IP address` s'affiche pour indiquer qu'aucune communication n'a été établie.

Une connexion entre CPUM Configurator et une carte CPUM doit être établie (Connection Check Status: Connected!) pour que d'autres opérations CPUM Configurator puissent être effectuées.

## 6.8 Modification de l'adresse IP d'un rack VM600 (carte CPUM)

Pour modifier l'adresse IP d'un rack VM600(carte CPUM) au moyen de CPUM Configurator :

**1-** Cliquez sur le bouton **Configure IP** pour afficher les informations d'adresse IP et de masque de réseau (masque de sous-réseau) configurées pour les interfaces Ethernet de la carte CPUM.

La boîte de dialogue **CPUM TCP-IP Configuration** qui apparaît affiche les valeurs **IP Address** et **Netmask** configurées pour chaque interface réseau (contrôleur Ethernet principal et contrôleur Ethernet secondaire) à partir de la carte CPUM. En outre, la valeur **Force 10 Mbps** indique si le débit de transfert de données de l'interface réseau est limité.

**2-** Pour modifier une adresse IP, dans la zone de texte **IP Address** appropriée, entrez l'adresse IP requise pour le contrôleur Ethernet principal ou secondaire de la carte CPUM en notation décimale à point.

Pour modifier un masque de sous-réseau, dans la zone de texte **Netmask** appropriée, entrez le masque de sous-réseau requis pour le contrôleur Ethernet principal ou secondaire de la carte CPUM en notation décimale à point.

Pour modifier le débit de transfert de données de l'interface réseau pour le contrôleur Ethernet principal de la carte CPUM, activez ou désactivez la case à cocher **Force 10 Mbps** selon le cas : la sélection de **Force 10 Mbps** limite le taux de transfert de données à 10 Mbps.

---

**NOTE :** La case à cocher **Secondary Ethernet Controller** doit être activée pour modifier l'adresse IP et le masque de sous-réseau de l'interface réseau secondaire. Il n'est pas nécessaire de limiter le débit de transfert de données de la deuxième interface de réseau secondaire, raison pour laquelle la case à cocher **Force 10 Mbps** n'est pas disponible (estompée).

---

**3-** Cliquez sur **OK** pour transférer les nouveaux paramètres d'adresse IP à la carte CPUM.

Les valeurs affichées dans la boîte de dialogue **CPUM TCP-IP Configuration** correspondent aux informations stockées dans le fichier `net.cfg` de la carte CPUM. Entrez une nouvelle adresse IP ou un nouveau masque de sous-réseau (masque de réseau) dans la boîte de dialogue CPUM TCP-IP Configuration et cliquez sur OK pour enregistrer les modifications apportées au fichier.

Cependant, pour que les modifications apportées au fichier `net.cfg` entrent en vigueur, la carte CPUM doit être redémarrée.

**4-** Comme vous y invite l'instruction qui s'affiche dans la zone de texte Action status, cliquez sur **Reboot** pour redémarrer la carte CPUM.

Vous pouvez également retirer et insérer la carte CPUM, ou éteindre puis rallumer l'alimentation du rack VM600, afin de redémarrer la carte CPUM.

**5-** La carte CPUM redémarre en utilisant les nouveaux paramètres d'adresse IP.

---

**NOTE :** Après le redémarrage de la carte CPUM, il est nécessaire de rétablir les communications avec la carte CPUM en utilisant la nouvelle adresse IP affichée dans la zone de texte **IP Address**.

---

## 6.9 Téléchargement de tous les fichiers de configuration à partir d'une carte CPUM

CPUM Configurator permet de télécharger simultanément tous les fichiers de configuration d'une carte CPUM (autrement dit, la configuration utilisateur (UC) complète).

Pour télécharger tous les fichiers de configuration d'une carte CPUM :

**1-** Sous **User Configuration**, cliquez sur le bouton **Save UC from CPUM**.

La boîte de dialogue **Create New CPUM UC** apparaît.

**2-** Sous **Path**, cliquez sur le bouton Parcourir (**...**) et utilisez la boîte de dialogue qui apparaît pour naviguer dans les dossiers de l'ordinateur et sélectionner un dossier dans lequel enregistrer les fichiers de configuration. Cliquez sur **OK** pour continuer.

Sous **UC tag**, saisissez le nom de la balise à utiliser dans le fichier `uc.cfg` de résumé généré par le téléchargement. Ce nom de balise s'utilise aussi pour le sous-dossier dans lequel les fichiers de configuration sont enregistrés.

Sous **Description**, acceptez la valeur par défaut ou entrez une description pour les fichiers de configuration.

Sous **FW version**, acceptez la valeur par défaut ou entrez une description pour le microprogramme exécuté sur la carte CPUM.

**3-** Cliquez ensuite sur **Create** pour télécharger tous les fichiers de configuration de la carte CPUM et créer une copie locale de la configuration utilisateur sur l'ordinateur.

La structure des répertoires pour les fichiers de configuration téléchargés est la suivante :

```
\UC tag\uc.cfg
\UC tag\etc\net.cfg
\UC tag\etc\cpum\launcher\launcher.cfg
\UC tag\etc\cpum\mbsrv\modbusDefault.cfg
\UC tag\etc\cpum\xmcsrv\xmcnn.xml
\UC tag\etc\cpum\xmcsrv\xmcsrv.cfg
\UC tag\etc\ntp\ntp.conf
```

Les fichiers de configuration téléchargés à partir d'une carte CPUM peuvent être modifiés localement sur un ordinateur puis chargés sur la carte, afin de modifier la configuration qui s'exécute sur la carte.

L'opération de téléchargement des fichiers de configuration peut également être utilisée comme sauvegarde de la configuration de la carte CPUM.

## 6.10 Chargement de tous les fichiers de configuration sur d'une carte CPUM

CPUM Configurator permet de charger simultanément tous les fichiers de configuration sur une carte CPUM (autrement dit, une configuration utilisateur (UC) complète).

Pour charger tous les fichiers de configuration sur une carte CPUM :

**1-** Sous **User Configuration**, cliquez sur le bouton **Send UC to CPUM**.

La boîte de dialogue **Open** apparaît.

**2-** Utilisez la boîte de dialogue pour naviguer dans les dossiers de l'ordinateur et sélectionner le fichier `uc.cfg` de résumé correspondant à la configuration à charger.

**3-** Cliquez ensuite sur **Send UC** pour charger tous les fichiers de configuration à partir de la copie locale de la configuration utilisateur complète stockée sur l'ordinateur sur la carte CPUM.

**4-** Comme vous y invite l'instruction qui s'affiche dans la zone de texte Action status, cliquez sur **Reboot** pour redémarrer la carte CPUM, ce qui oblige la carte à utiliser la nouvelle configuration.

Vous pouvez également retirer et insérer la carte CPUM, ou éteindre puis rallumer l'alimentation du rack VM600, afin de redémarrer la carte CPUM.

La structure de répertoires requise pour les fichiers de configuration chargés est la même que la structure créée lors du téléchargement de tous les fichiers de configuration. Voir la section 6.9 Téléchargement de tous les fichiers de configuration à partir d'une carte CPUM.

Les fichiers de configuration téléchargés à partir d'une carte CPUM peuvent être modifiés localement sur un ordinateur puis chargés sur la carte, afin de modifier la configuration qui s'exécute sur la carte.

L'opération de chargement des fichiers de configuration peut également être utilisée pour restaurer la configuration d'une carte CPUM.

---

**NOTE :** Le fichier `net.cfg` contient les informations d'interface IP (adressage IP) d'une carte CPUM exécutant la version 075 ou ultérieure du microprogramme. Lorsqu'une carte CPUM démarre, elle lit le fichier `net.cfg` afin d'apprendre ses paramètres d'interface réseau.
Par conséquent, si un fichier `net.cfg` possédant des paramètres différents est chargé sur une carte CPUM et si la carte CPUM est redémarrée, il est nécessaire de rétablir les communications avec la carte CPUM en utilisant la nouvelle adresse IP.

---

## 6.11 Chargement de fichiers de configuration individuels sur une carte CPUM

Avec CPUM Configurator, certains fichiers de configuration d'une carte CPUM peuvent également être chargés individuellement.

### 6.11.1 Chargement d'un fichier de configuration NTP sur une carte CPUM

Pour charger un fichier de configuration NTP sur une carte CPUM :

**1-** Sous **Upload individual configuration files to CPUM**, cliquez sur le bouton **NTP**.

La boîte de dialogue **Open** apparaît.

**2-** Utilisez la boîte de dialogue pour parcourir les dossiers sur l'ordinateur et sélectionner le fichier `ntp.conf` à charger.

**3-** Cliquez ensuite sur **Send** pour charger le fichier de configuration sur la carte CPUM.

---

**NOTE :** La carte CPUM doit être redémarrée pour que cette modification de configuration s'applique au processus du serveur de synchronisation réseau en cours d'exécution.

---

**4-** Le message qui s'affiche dans la zone de texte **Action status** indique le résultat de l'opération.

**5-** Comme vous y invite l'instruction qui s'affiche dans la zone de texte Action status, cliquez sur **Reboot** pour redémarrer la carte CPUM, ce qui oblige la carte à utiliser la nouvelle configuration.

Vous pouvez également retirer et insérer la carte CPUM, ou éteindre puis rallumer l'alimentation du rack VM600, afin de redémarrer la carte CPUM.

### 6.11.2 Chargement d'un fichier de configuration Modbus sur une carte CPUM

Pour charger un fichier de configuration Modbus sur une carte CPUM :

**1-** Sous **Upload individual configuration files to CPUM**, cliquez sur le bouton **Modbus**.

La boîte de dialogue **Open** apparaît.

**2-** Utilisez la boîte de dialogue pour parcourir les dossiers sur l'ordinateur et sélectionner le fichier `modbusDefault.cfg` à charger.

**3-** Cliquez ensuite sur **Configure** pour charger le fichier de configuration sur la carte CPUM.

**NOTE :** Lorsque le fichier est chargé, il est vérifié (analysé) par la carte CPUM pour s'assurer qu'il ne contient pas d'erreurs et que la configuration est valide.

**4-** Le message qui s'affiche dans la zone de texte **Action status** indique le résultat de l'opération.

La carte CPUM applique automatiquement cette modification de configuration au processus Modbus en cours d'exécution (aucun redémarrage n'est requis).

### 6.11.3 Chargement d'un fichier de configuration XMx16 sur une carte CPUM

**NOTE :** Dans l'interface utilisateur de CPUM Configurator, le bouton **XMx** est indisponible (grisé) lorsque l'utilisateur est connecté avec le nom d'utilisateur et le mot de passe par défaut (`user` et `config`, respectivement) car ce bouton est destiné à une fonction spéciale réservée à une utilisation en usine. Il est décrit ci-dessous à titre d'information uniquement.

Certaines applications de surveillance des machines exigent qu'une carte VM600XMx16 (XMC16, XMV16 ou XMVS16) prenne en charge la communication avec plusieurs serveurs en même temps. Par exemple, pour permettre à un système de contrôle et à un système de surveillance d'état fonctionnant en parallèle de « partager » les données de mesure d'une carte XMx16.

Les deux serveurs utilisés avec une carte XMx16 partagée peuvent être soit deux modules logiciels VibroSight Server exécutés sur un ordinateur hôte ou des ordinateurs, soit un module logiciel VibroSight Server s'exécutant sur un ordinateur hôte et un processus de serveur VibroSight s'exécutant sur une carte CPUM.

Le processus serveur VibroSight exécuté sur une carte CPUM nécessite une copie de la configuration de la carte XMx16 partagée afin de pouvoir communiquer avec elle. La carte CPUM peut obtenir la configuration de la carte XMx16 de l'une des deux façons suivantes :

- En utilisant CPUM Configurator pour charger la configuration de la carte XMx16 directement à partir de la carte XMx16 via le bus VME qui se trouve sur le panneau arrière du rack VM600.
- En utilisant CPUM Configurator pour charger la configuration de la carte XMx16 à partir d'un fichier de configuration de périphérique individuel (`*.xml`) pour une carte XMx16 exportée à partir du module VibroSight Configurator du logiciel VibroSight®.

Pour éviter les conflits, un seul des deux serveurs peut avoir un accès en lecture et en écriture à la carte VM60 XMx16 partagée (l'autre serveur aura un accès en lecture seule).

Le serveur avec accès en lecture et écriture est le « maître » et dans une application type, il s'agirait du système le plus important, par exemple, un système de contrôle. Cet accès est configuré à l'aide de VibroSight Configurator.

Pour charger un fichier de configuration de carte XMx16 sur une carte CPUM :

**1-** Sous **Upload individual configuration files to CPUM**, cliquez sur le bouton **XMx**.

La boîte de dialogue **XMCSRV Configuration** qui apparaît permet de spécifier les configurations de carte XMx16 pour deux cartes XMx16 maximum.

**2-** Pour chaque configuration de carte XMx16 « partagée » avec le processus serveur VibroSight exécuté sur la carte CPUM, sous **XMx Card 1** et **XMx Card 2** :

- Saisissez l'adresse IP (**IP Address**) et le numéro de logement du rack VM600 (**Slot**) pour la carte.
- Sélectionnez **Force pairing** si le processus serveur VibroSight qui s'exécute sur la carte CPUM doit être le « maître » de la carte XMx16 (c'est-à-dire permettre l'accès en lecture et en écriture à la carte XMx16 par la carte CPUM).
- Sous **XMC board configuration** :

  Cliquez sur **get configuration from Xmx card** pour charger la configuration de la carte XMx16 directement à partir de la carte XMx16 via le bus VME qui se trouve sur le panneau arrière du rack VM600 ou

  Cliquez sur **set configuration from file**, cliquez sur le bouton Parcourir (**...**) et utilisez la boîte de dialogue qui apparaît pour naviguer dans les dossiers de l'ordinateur et sélectionner un fichier de configuration de périphérique individuel (`xmcnn.xml`) pour charger la configuration de la carte XMx16 exportée à partir du module VibroSight Configurator du logiciel VibroSight®.

**3-** Cliquez ensuite sur **Configure** pour charger le fichier de configuration sur la carte CPUM.

---

**NOTE :** Vous pouvez aussi utiliser le bouton **Import** pour compléter la boîte de dialogue **XMCSRV Configuration** à partir des informations enregistrées dans un fichier `xmcsrv.cfg` (par exemple, à partir d'une configuration utilisateur complète précédemment enregistrée).

---

**4-** Le message qui s'affiche dans la zone de texte **Action status** indique le résultat de l'opération.

La carte CPUM applique automatiquement cette modification de configuration au processus en cours d'exécution sur la carte XMx16 (aucun redémarrage n'est requis).

## 6.12 Modification de la configuration d'une carte CPUM

Avec CPUM Configurator, la méthode recommandée pour modifier la configuration d'une carte CPUM consiste à :

**1-** Utiliser le bouton **Save UC from CPUM** pour télécharger une copie de la configuration utilisateur (UC) complète de la carte CPUM sur un ordinateur.

Voir la section 6.9 Téléchargement de tous les fichiers de configuration à partir d'une carte CPUM.

**2-** Modifier la copie locale de la configuration sur l'ordinateur à l'aide d'un éditeur de texte approprié.

3- Utiliser le bouton **Send UC to CPUM** pour charger la copie modifiée de la configuration utilisateur de l'ordinateur vers la carte CPUM.

Voir la section 6.10 Chargement de tous les fichiers de configuration sur d'une carte CPUM.

## 6.13 Modification des options de CPUM Configurator

Le logiciel CPUM Configurator a un comportement par défaut qui peut être modifié au moyen des paramètres des options :

1- Cliquez sur le bouton **Options** pour afficher les paramètres configurables par l'utilisateur pour CPUM Configurator.

La boîte de dialogue Options qui s'affiche indique le nom d'utilisateur et le mot de passe utilisés pour se connecter à CPUM Configurator et établir une connexion avec une carte CPUM.

2- Sélectionnez **Login Automatically** et cliquez sur **OK** pour que CPUM Configurator mémorise et utilise le mot de passe pour se connecter automatiquement au prochain démarrage du logiciel (pas de fenêtre de connexion).

## 6.14 Redémarrage d'une carte CPUM

Cliquez sur le bouton **Reboot** pour forcer la carte CPUM à redémarrer.

(Ceci équivaut à éteindre et rallumer l'alimentation de la carte CPUM.)

Si CPUM Configurator a effectué un opération qui nécessite un redémarrage, telle que la modification d'une adresse IP ou d'une configuration NTP (Network Time Protocol) :

- Un message apparaît dans la zone de texte Action status.
- Un message s'affiche au-dessus du bouton Reboot (**Reboot is required to apply the changes**) pour rappeler que le redémarrage de la carte CPUM est nécessaire.

## 6.15 Mode ATP

**NOTE :** Dans l'interface utilisateur de CPUM Configurator, le bouton **ATP mode** est indisponible (grisé) lorsque l'utilisateur est connecté avec le nom d'utilisateur et le mot de passe par défaut (`user` et `config`, respectivement) car ce bouton est destiné à une fonction spéciale réservée à une utilisation en usine. Il est décrit ci-dessous à titre d'information uniquement.

Le bouton du mode ATP place la carte CPUM dans un mode de procédure de test d'acceptation (ATP, acceptance test procedure) qui permet le test de la carte (réservé à l'utilisation en usine de Meggitt uniquement).

Cependant, si une carte CPUM est placée par inadvertance en mode ATP, il suffit d'appuyer sur le bouton ALARM RESET du panneau avant de la carte CPUM pour quitter le mode ATP et revenir au fonctionnement normal avec la configuration actuelle.

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# 7 ÉMULATION DE TERMINAL

La gestion de racks VM600 en réseau avec une carte CPUM exécutant la version 075 ou ultérieure du microprogramme peut être effectuée au moyen d'un émulateur de terminal et de Midnight Commander.

Ce chapitre du manuel utilise le programme d'émulation de terminal PuTTY pour démontrer l'utilisation de l'émulation de terminal avec une carte CPUM, mais d'autres émulateurs de terminal peuvent également être utilisés, par exemple, le programme HyperTerminal ou un terminal vidéo VT100.

**NOTE :** Midnight Commander est disponible sur les cartes CPUM exécutant la version 074 ou ultérieure du microprogramme.

Pour identifier la version du microprogramme d'une carte CPUM, voir la section 3.3 Identification de la version du microprogramme d'une carte CPUM.

Pour travailler avec les cartes CPUM exécutant la version 073 ou antérieure du microprogramme, reportez-vous au *Manuel de mise en réseau du VM600, édition 6 (ou antérieure)*.

## 7.1 À propos de PuTTY

PuTTY est un programme gratuit d'émulation de terminal open-source qui peut être utilisé pour communiquer avec une carte CPUM dans un rack VM600. Le programme PuTTY prend en charge le protocole Telnet (client Telnet) qui permet la communication avec une carte CPUM (client Telnet) via une liaison Ethernet. Il prend également en charge les protocoles de communication série qui permettent la communication avec une carte CPUM via une liaison série.

**NOTE :** CPUM Configurator est compatible avec les cartes CPUM exécutant la version 075 ou ultérieure du microprogramme.

Le logiciel Putty est disponible sous la forme d'un fichier programme individuel (`*.exe`) ou sous forme de programme d'installation, téléchargeable à l'adresse :

- http://www.putty.org
- http://www.chiark.greenend.org.uk/~sgtatham/putty

Windows® et d'autres systèmes d'exploitation sont pris en charge.

## 7.2 À propos de Midnight Commander

Midnight Commander est un gestionnaire de fichiers qui fournit une interface utilisateur textuelle permettant d'effectuer facilement des opérations de navigation dans les répertoires, de gestion de fichiers et d'édition de fichiers texte. L'interface principale se compose de deux panneaux qui affichent le système de fichiers. La sélection des fichiers s'effectue à l'aide des touches fléchées, la touche ENTRÉE s'utilise pour sélectionner les fichiers et les touches de fonction exécutent des opérations telles que renommer, éditer et copier des fichiers. Midnight Commander inclut également un éditeur (appelé mcedit) et assure la prise en charge de la souris.

Pour démarrer Midnight Commander, tapez `mc` (puis appuyez sur ENTRÉE) sur la ligne de commande lorsque vous êtes connecté à une carte CPUM dans une session d'émulation de terminal.

## 7.3 Démarrage d'une session d'émulation de terminal à l'aide de PuTTY et établissement de communications avec une carte CPUM

Pour démarrer une session d'émulation de terminal à l'aide de PuTTY et établir des communications avec une carte CPUM :

**1-** Double-cliquez sur l'icône PuTTY (raccourci) sur le bureau Windows.

Vous pouvez aussi accéder au fichier programme PuTTY (`putty.exe`) sur le disque dur local de l'ordinateur et double-cliquer dessus.

**2-** Démarrez PuTTY, sélectionnez une connexion **Telnet** et entrez l'adresse IP (**Nom d'hôte (ou adresse IP)**) de la carte CPUM, comme indiqué sur la Figure 7-1. (Telnet utilise toujours le numéro de port TCP 23 (**Port**).)

Vous pouvez aussi utiliser un éditeur de terminal différent qui prend en charge les connexions Telnet.

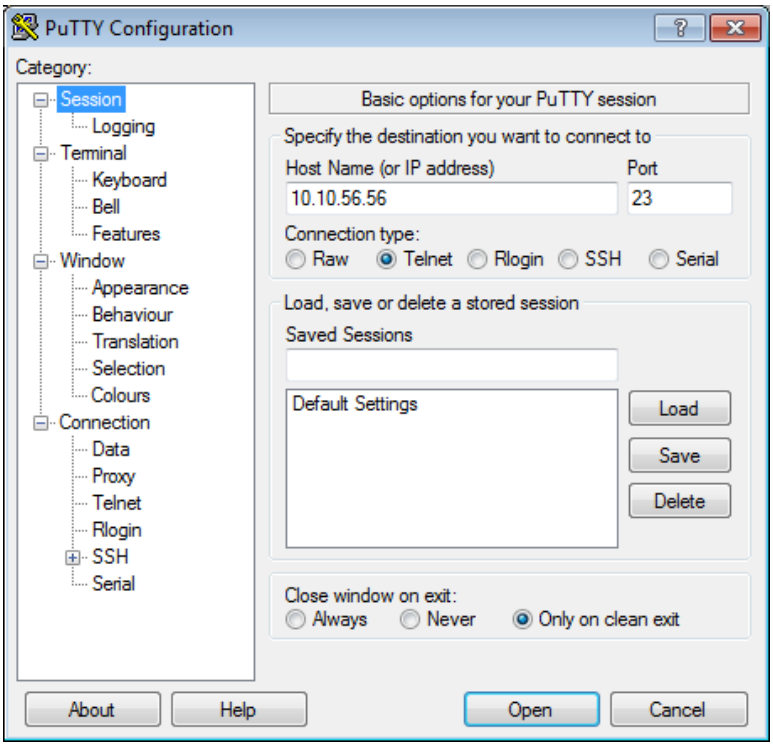



**Figure 7-1:** Utilisation du logiciel d'émulation de terminal PuTTY – sélection et configuration d'une connexion Telnet

**3-** Acceptez les options de session Telnet par défaut de PuTTY, comme indiqué sur la Figure 7-2.

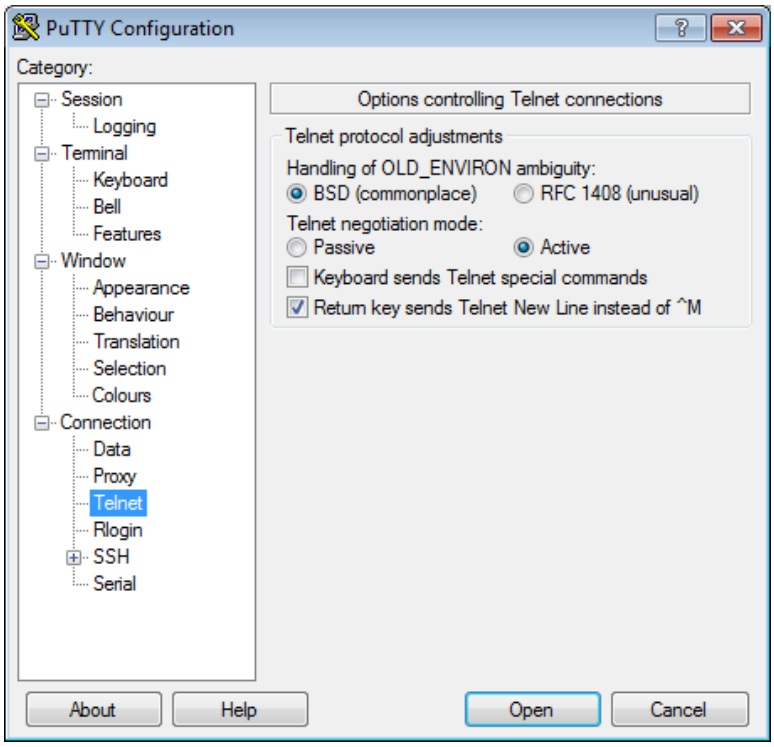



**Figure 7-2:** Utilisation du logiciel d'émulation de terminal PuTTY – options Telnet par défaut

**4-** Cliquez sur **Open** pour démarrer la session PuTTY, puis appuyez sur la touche ENTRÉE.

**5-** Lorsque vous êtes invité à vous connecter, tapez **user** (identifiant de connexion par défaut), puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
login : user
```

Lorsque vous êtes invité à saisir un mot de passe, tapez **config** (mot de passe par défaut), puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
password : config
```

L'invite de ligne de commande ( `/usr/user #` ) s'affiche lorsque la connexion est établie.

La ligne de commande du terminal et Midnight Commander sont utilisés pour effectuer toutes les opérations.

## 7.4 Arrêt d'une session d'émulation de terminal à l'aide de PuTTY

Pour fermer une session d'émulation de terminal à l'aide de PuTTY, procédez comme suit :

- Tapez `logout` (puis appuyez sur ENTRÉE) sur la ligne de commande du terminal.
- Vous pouvez aussi cliquer sur la case de **fermeture** de Windows (en haut à droite).

Ceci termine la session d'émulation de terminal et quitte PuTTY.

## 7.5 Aide sur l'utilisation de PuTTY

Pour obtenir de l'aide sur l'utilisation de PuTTY :

- Après avoir démarré PuTTY, cliquez sur le bouton **Help**.

Le manuel d'utilisation de PuTTY s'affiche.

## 7.6 Démarrage de Midnight Commander

Pour démarrer Midnight Commander :

- Tapez `mc` (puis appuyez sur ENTRÉE) sur la ligne de commande du terminal.

Midnight Commander démarre et son interface utilisateur s'affiche dans la fenêtre du terminal, comme indiqué sur la Figure 7-3.

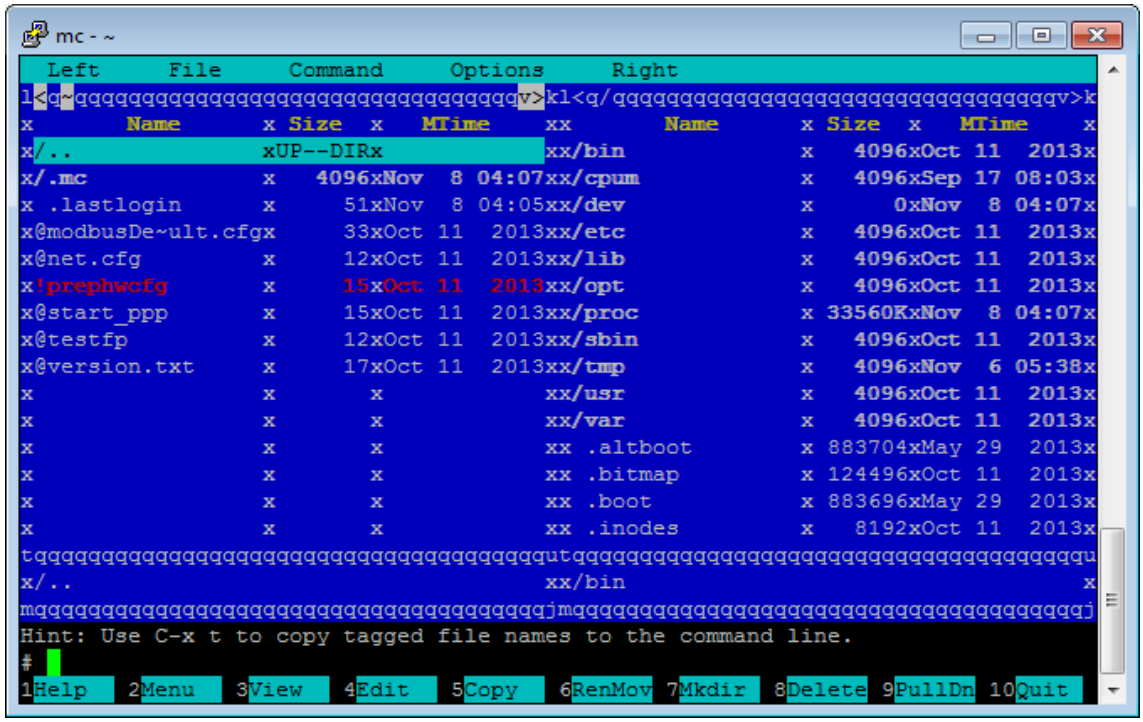



**Figure 7-3:** Interface utilisateur de Midnight Commander

La ligne de commande et Midnight Commander sont utilisés pour effectuer toutes les opérations.

## 7.7 Interface utilisateur de Midnight Commander

L'interface utilisateur de Midnight Commander comprend :

- Une ligne de menus (en haut) donnant accès aux commandes.
- Deux volets de répertoires (à gauche et à droite) qui affichent le système de fichiers.
- Une ligne de commande (# ) permettant d'entrer les commandes de Midnight Commander.

  Juste au-dessus de la ligne de commande se trouve une ligne d'astuces qui affiche des conseils aléatoires.
- Une ligne de touches de fonction (en bas) offrant des raccourcis clavier pour accéder aux commandes les plus fréquemment utilisées.

#### 7.7.0.1 Navigation dans Midnight Commander

Pour basculer entre les deux volets de répertoires (gauche ou droit), appuyez sur la touche tabulation pour passer à l'autre volet ou utilisez le pointeur pour cliquer dans le volet de répertoires requis.

Pour vous déplacer dans un volet de répertoires, utilisez les touches fléchées HAUT et BAS, ou utilisez le pointeur. Les touches Page précédente et Page suivante permettent de se déplacer vers le haut et le bas d'un long répertoire, écran par écran. Les touches ORIGINE et FIN permettent de passer directement au début ou à la fin d'un répertoire long.

Pour basculer vers un répertoire parent, placez-vous au début du répertoire (`/..`) et appuyez sur la touche ENTRÉE. (`/..` fait référence à un répertoire parent dans le système de fichiers Unix utilisé par une carte CPUM.)

Pour basculer vers un sous-répertoire, accédez au répertoire requis (`/directoryname`) et appuyez sur la touche ENTRÉE.

#### 7.7.0.2 Raccourcis clavier de Midnight Commander

Les touches de fonction (F1 à F10) donnent accès aux commandes les plus fréquemment utilisées.

Vous pouvez également utiliser le pointeur pour sélectionner une fonction en cliquant dessus sur la ligne des touches de fonction (dans la partie inférieure).

La fonction réellement exécutée peut différer selon le contexte, par exemple, lors de la navigation dans un répertoire, de l'utilisation du visualiseur de fichiers ou de l'utilisation de l'éditeur. La fonction contextuelle s'affiche toujours sur la ligne des touches de fonction (dans la partie inférieure).

Si aucune touche de fonction physique n'est disponible, la fonction reste malgré tout disponible par le biais d'une séquence de touches d'échappement. Appuyer sur la touche ÉCHAP et taper ensuite **1**, par exemple, équivaut à appuyer sur la touche de fonction F1. (Appuyer sur la touche ÉCHAP et taper ensuite **0** équivaut à appuyer sur la touche de fonction F10.)

Pour basculer entre la console du terminal (invite de ligne de commande) et Midnight Commander sans quitter mc, appuyez sur la touche CTRL, puis tapez **o** (c'est-à-dire, la lettre minuscule).

Les raccourcis des commandes de menu (partie supérieure) sont affichés dans les menus.

#### 7.7.0.3 Menus de Midnight Commander

Les menus donnent accès à toutes les commandes disponibles dans Midnight Commander.

Menu **Left** – commandes permettant de travailler avec le volet de répertoires gauche. Par exemple, pour modifier les informations affichées dans le volet.

Menu **File** – commandes permettant de travailler sur les fichiers. Par exemple, pour afficher ou modifier le fichier sélectionné.

Menu **Commands** – autres commandes.

Menu **Options** – commandes permettant de configurer l'apparence et le comportement de Midnight Commander.

Menu **Right** – commandes permettant de travailler avec le volet de répertoires droit. Par exemple, pour modifier les informations affichées dans le volet.

Pour afficher les commandes d'un menu, appuyez sur la touche F9 (PullDn) pour activer les menus et utilisez les touches fléchées pour naviguer dans les menus et sélectionner une commande. Appuyez sur ENTRÉE pour exécuter la commande sélectionnée. Pour désactiver les menus, appuyez sur la touche ÉCHAP, puis appuyez sur une touche fléchée.

Vous pouvez également utiliser le pointeur pour afficher, sélectionner et/ou exécuter une commande de menu en cliquant dessus dans le menu (partie supérieure).

## 7.8 Fermeture de Midnight Commander

Pour fermer Midnight Commander :

- Appuyez sur la touche F10 (Quitter) ou utilisez le pointeur pour cliquer dessus.

Cette opération entraîne la fermeture de Midnight Commander (c'est-à-dire, du processus mc en cours d'exécution sur la carte CPUM).

## 7.9 Aide sur l'utilisation de Midnight Commander

Pour obtenir de l'aide sur l'utilisation de Midnight Commander :

- Appuyez sur la touche F1 (Aide) ou utilisez le pointeur pour cliquer sur 1 Aide.

L'aide de Midnight Commander s'affiche.

## 7.10 Modification de l'adresse IP d'un rack VM600 (carte CPUM)

Pour modifier l'adresse IP d'un rack VM600 (carte CPUM) au moyen de PuTTY et Midnight Commander :

**1-** Dans un volet de répertoires de Midnight Commander, accédez au fichier `net.cfg`, comme indiqué sur la Figure 7-4 (à gauche).

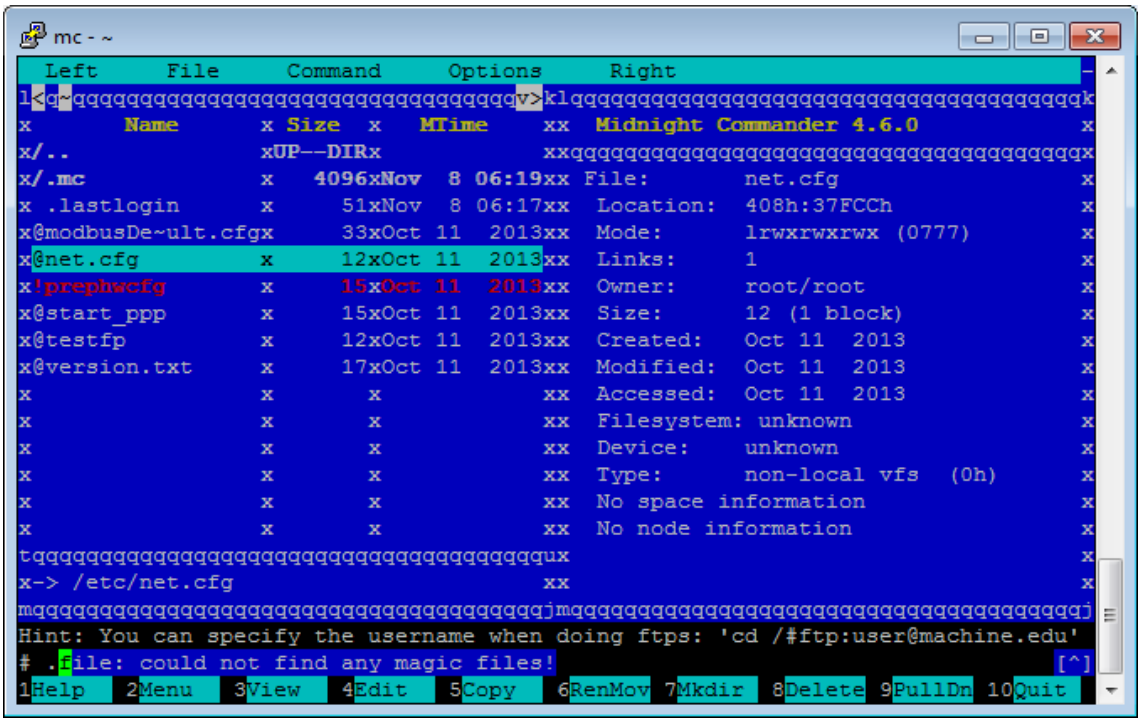



**Figure 7-4:** Utilisation de Midnight Commander pour sélectionner un fichier

Le chemin d'accès au fichier `net.cfg` est `/usr/user/net.cfg`

**2-** Pour modifier le fichier à l'aide de Midnight Commander, appuyez sur la touche F4 (Modifier) ou utilisez le pointeur pour cliquer sur 4 Modifier.
Le fichier sélectionné (`net.cfg`) est ouvert dans l'éditeur de Midnight Commander, comme indiqué sur la Figure 7-5.

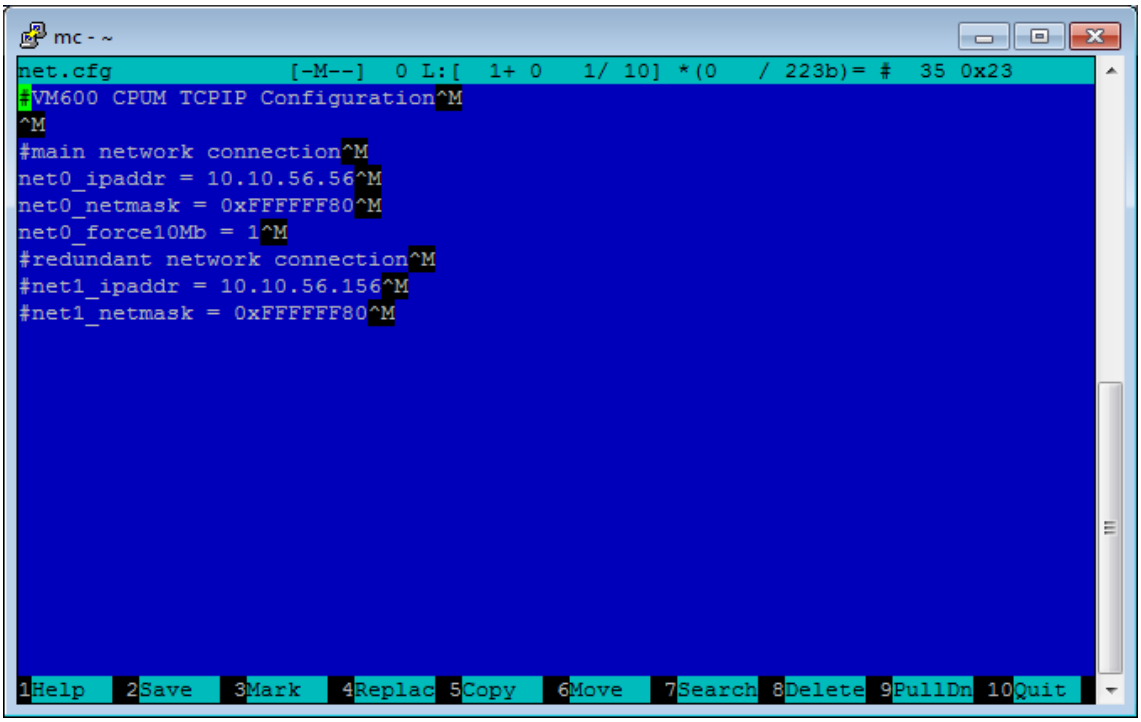



**Figure 7-5:** Utilisation de Midnight Commander pour modifier un fichier

Le fichier `net.cfg` affiche l'adresse IP (**netx_ipaddr**) et le masque de sous-réseau (**netx_netmask**) actuellement configurés pour les interfaces Ethernet de la carte CPUM (interface réseau principale (ETH1) et interface réseau secondaire (ETH2)). En outre, la valeur **net0_force10Mb** indique si le débit de transfert de données de l'interface réseau est limité. Voir la section 3.4.2 À propos du fichier net.cfg pour plus d'informations.

**3-** Pour modifier une adresse IP, modifiez la ligne **netx_ipaddr** appropriée et tapez l'adresse IP requise en notation décimale à point.

Pour modifier un masque de sous-réseau, modifiez la ligne **netx_netmask** appropriée et tapez le masque de sous-réseau requis en notation hexadécimale.

Pour modifier le débit de transfert de données de l'interface réseau principale (ETH1), sous `#main network connection`, modifiez la ligne **net0_force10Mb** en fonction des besoins : **net0_force10Mb** = 1 limite le taux de transfert de données à 10 Mbps et **net0_force10Mb** = 0 ne limite pas le taux de transfert de données.

---

**NOTE :** Les caractères ^M à la fin de chaque ligne correspondent à un retour chariot (0x0D) et ne doivent pas être supprimés.
Seul le clavier de l'ordinateur principal doit être utilisé lors de l'édition de fichiers avec Midnight Commander : n'utilisez pas de pavé numérique.

---

**4-** Pour enregistrer le fichier modifié, appuyez sur la touche F2 (Enregistrer) ou utilisez le pointeur pour cliquer sur 2 Enregistrer.

Lorsque vous y êtes invité, sélectionnez **Save** pour enregistrer le fichier et les modifications apportées à la carte CPUM. La touche F10 (Quitter) permet de quitter le fichier sans enregistrer les modifications.

Cependant, pour que les modifications apportées au fichier `net.cfg` entrent en vigueur, la carte CPUM doit être redémarrée.

**5-** Pour fermer (quitter) Midnight Commander et revenir à la console du terminal (invite de ligne de commande), appuyez sur la touche F10 (Quitter) ou utilisez le pointeur pour cliquer sur 10 Quitter.

**6-** Pour redémarrer la carte CPUM, à l'invite de commande ( `/usr/user #` ), tapez **shutdown -f**, puis appuyez sur ENTRÉE pour continuer. Par exemple :

```
/usr/user # shutdown –f
```

La commande shutdown arrête et redémarre la carte CPUM.

Vous pouvez également retirer et insérer la carte CPUM, ou éteindre puis rallumer l'alimentation du rack VM600, afin de redémarrer la carte CPUM.

**7-** La carte CPUM redémarre en utilisant les nouveaux paramètres d'adresse IP.

---

**NOTE :** Après le redémarrage de la carte CPUM, il est nécessaire de rétablir les communications avec la carte CPUM en utilisant la nouvelle adresse IP affichée dans la zone de texte **IP Address**.

---

## 7.11 Affichage du contenu d'un fichier

Pour afficher le contenu d'un fichier texte au moyen de PuTTY et Midnight Commander :

**1-** Dans un volet de répertoires de Midnight Commander, accédez à un fichier (par exemple, `version.txt`), comme indiqué sur la Figure 7-6 (à gauche).

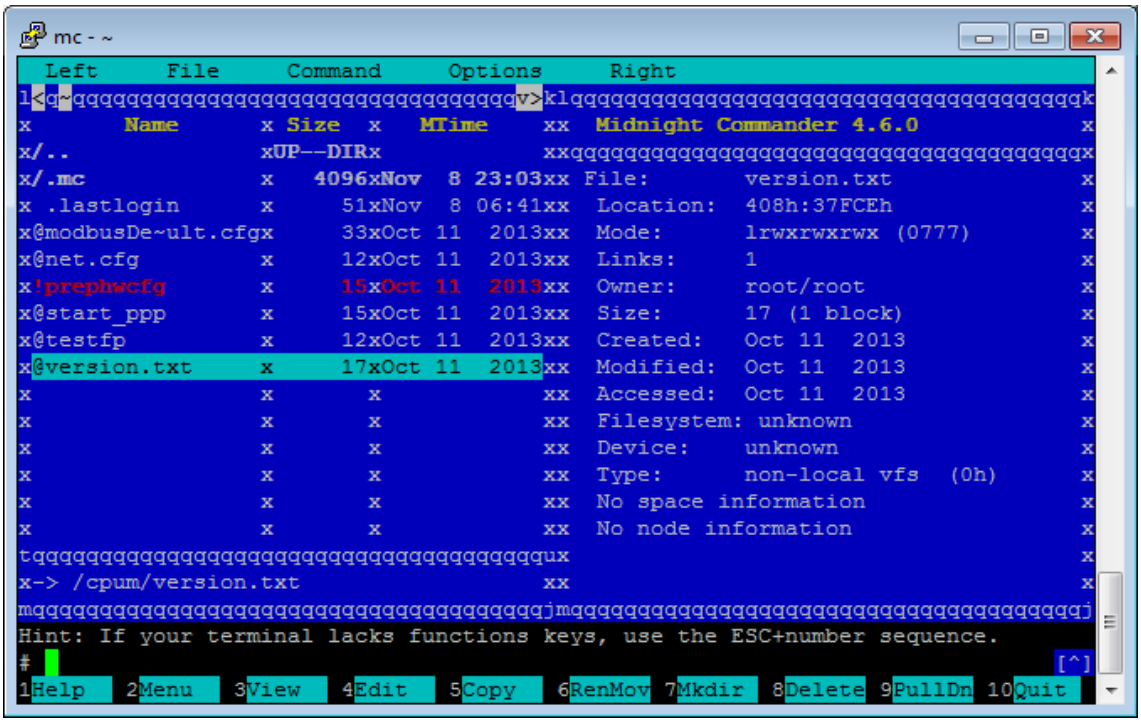



**Figure 7-6:** Utilisation de Midnight Commander pour sélectionner un fichier

Le chemin d'accès au fichier `version.txt` est `/usr/user/version.txt`

**2-** Pour afficher le fichier à l'aide de Midnight Commander, appuyez sur la touche F3 (Afficher) ou utilisez le pointeur pour cliquer sur 3 Afficher.
Le fichier sélectionné (`version.txt`) s'affiche dans Midnight Commander, comme indiqué sur la Figure 7-7.

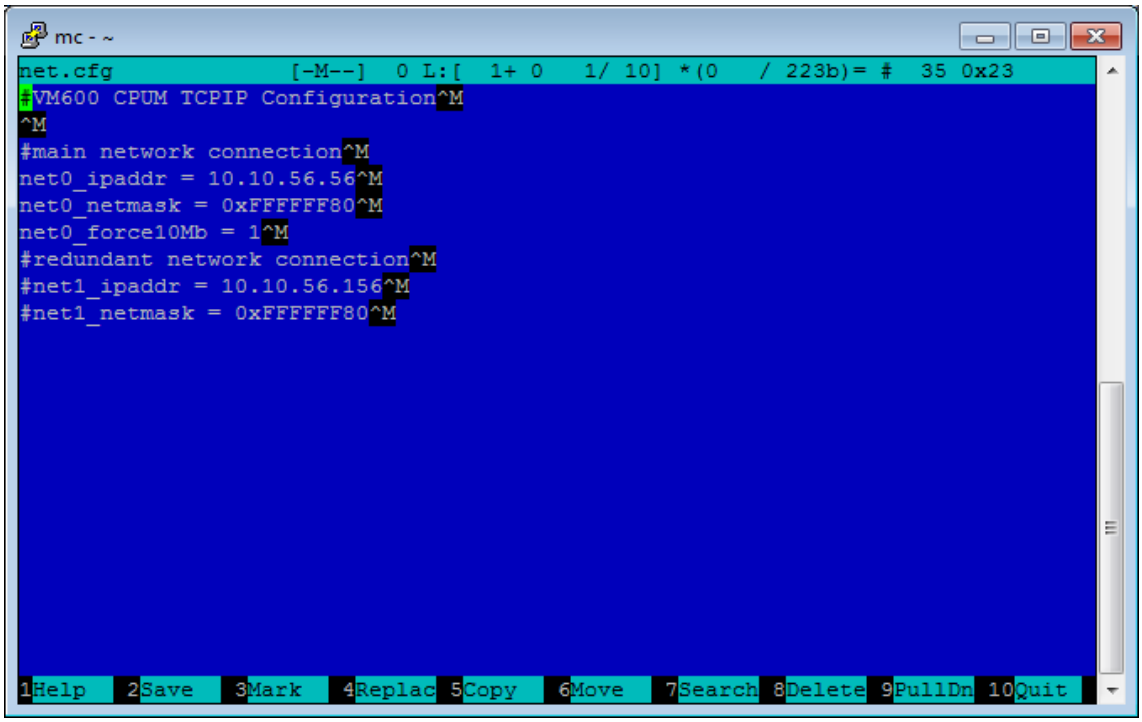



**Figure 7-7:** Utilisation de Midnight Commander pour afficher un fichier

Le fichier version.txt affiche les informations de version d'une carte CPUM. Voir la section 3.3.1 À propos du fichier version.txt pour plus d'informations.

**3-** Pour fermer le fichier affiché, appuyez sur la touche F10 (Quitter) ou utilisez le pointeur pour cliquer sur 10 Quitter.

**4-** Pour fermer (quitter) Midnight Commander et revenir à la console du terminal (invite de ligne de commande), appuyez sur la touche F10 (Quitter) ou utilisez le pointeur pour cliquer sur 10 Quitter.

## 7.12 Communications série avec un rack VM600

Toute opération ou tâche pouvant être exécutée dans une session d'émulation de terminal peut également être effectuée dans une session de communication série (voir la section 3 Communications série avec un rack VM600).

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# 8 CONFIGURATION D'UNE CONNEXION MODBUS (LOGICIEL CPUM VERSION 067 OU ANTÉRIEURE)

## 8.1 Introduction

Ce chapitre décrit l'implémentation de l'interface logicielle Modbus pour les racks VM600 tel qu'utilisée par les cartes CPUM exécutant la version logicielle 067 ou antérieure.

Presque tous les registres Modbus sont disponibles pour les cartes CPUM exécutant la version logicielle 067 (ou antérieure). Les tableaux de l'Annexe A : Définitions de registre Modbus MPC4 et de l'Annexe B : Définitions de registre Modbus AMC8 contiennent une colonne (CPUM < 71) indiquant quels sont les registres Modbus disponibles dans la version logicielle précise d'un CPUM.

**NOTE :** Une implémentation plus souple du serveur Modbus est possible pour les cartes CPUM exécutant la version logicielle 071 ou ultérieure (voir 9 Configuration d'une connexion Modbus (logiciel CPUM version 071 ou ultérieure)).

L'interface logicielle Modbus fournit un serveur Modbus pour un rack VM600 qui communique avec les cartes VM600 suivantes :

- MPC4
- AMC8.

Les données transmises sont composées de valeurs en temps réel (niveau de vibration, pression etc.), et d'informations sur l'état et la configuration (alarmes). Ces données peuvent être utilisées par tout système externe, comme un système de commande distribué, aux fins de contrôle des machines.

Voir l'Annexe A : Définitions de registre Modbus MPC4 pour des informations détaillées sur le mappage du registre Modbus de la carte MPC4 et à l'Annexe B : Définitions de registre Modbus AMC8 pour des informations détaillées sur le mappage du registre Modbus de la carte AMC8.

## 8.2 Modbus

Modbus est apparu dans les années 70 sous forme d'un protocole de communication série publié par Modicon pour être utilisé avec des automates programmables industriels (API). Sur un réseau Modbus, les automates communiquent à l'aide d'une technologie maître-esclave, par laquelle un seul périphérique (le maître) peut lancer des requêtes (transactions). Les autres périphériques du réseau (esclaves) répondent en fournissant les données demandées au maître, ou en appliquant les mesures exigées dans la requête.

Depuis, Modbus a évolué de façon à devenir l'une des normes les plus largement utilisées dans les domaines de l'automatisation et du contrôle industriels. Il est devenu un protocole de message de la couche application qui permet d'établir une communication client-serveur entre des périphériques connectés sur différents types de bus ou de réseaux. Cependant, il reste principalement utilisé pour échanger des données dans le domaine de l'automatisation.

Aujourd'hui, Modbus est un protocole client-serveur basé sur des transactions (voir la Figure 8-1), qui englobe :

- Une requête envoyée par le client
- Une réponse envoyée par le serveur.

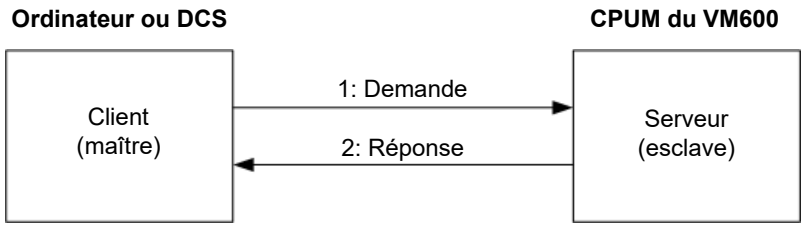



**Figure 8-1:** Une transaction Modbus

Dans ce modèle de communication Modbus client-serveur, la CPUM du VM600 est le serveur Modbus qui répond aux demandes de transaction externes provenant de l'équipement du client, comme un ordinateur ou un système de commande distribué (DCS).

Modbus est un protocole de requête/réponse qui offre des services spécifiés par des codes de fonction. Les codes de fonction Modbus sont des éléments des unités de données de protocole (PDU) de requête/réponse de Modbus. Ils sont présentés en détail dans la section 8.4 Description du protocole Modbus RTU.

Des versions du protocole Modbus existent pour les interfaces de communication série (le protocole Modbus RTU) et pour Ethernet (le protocole Modbus TCP).

Veuillez vous rendre sur le site Internet de l'entreprise Modbus pour accéder aux informations les plus récentes :
http://www.modbus.org/specs.php

## 8.3 Communication

La communication s'effectue de deux façons :

- Via une liaison série utilisant le protocole Modbus RTU standard de Modicon
  et/ou
- Via une liaison Ethernet utilisant le protocole Modbus TCP standard de Modicon.

### 8.3.1 Modbus RTU

Le protocole Modbus RTU peut être utilisé avec des interfaces série RS-232 et RS-485. Le champ d'adresse d'une trame de message Modbus RTU contient huit bits de sorte que les adresses valides d'un périphérique esclave sont comprises dans une fourchette de 1 à 247 (en décimal). (L'adresse 00 est réservée à l'envoi de transactions à tous les périphériques serveur du rseau, mais n'est pas prise en charge dans le cadre de l'implémentation Modbus de Meggitt.)

En pratique, le nombre maximum de systèmes ou périphériques pouvant être gérés est limité par la spécification RS-485. Elle indique que le bus peut gérer jusqu'à 32 « charges unitaires », de sorte que le nombre maximum de périphériques pouvant être connectés dépend de la charge que représente chaque périphérique connecté pour le système.

La liaison série peut être en semi-duplex ou en duplex-intégral avec des vitesses de transmission sélectionnables de 1200 à 19200 bit/s. La plage des paramètres de communication pris en charge est indiquée dans le Tableau 8-1. La configuration des paramètres de communication par défaut du VM600 est disponible dans la section 8.4.2 Paramètres de communication du VM600.

**Tableau 8-1 :** : Paramètres de communication RTU pris en charge par le VM600

| Débit en bauds | Parité | Adresse esclave |
|---|---|---|
| 600, 1200, 4800, 9600, 19200, 38400, 57600, 115200 | Even, odd ou none (Impaire, paire ou aucune) | 1 à 247 |

**NOTE :** Lorsque la carte CPUM contient un module de communication RS-485/RS-422 optionnel (permettant ainsi de communiquer via les paires de connecteurs « A » et « B » sur le IOCN), seule une transmission en duplex intégral est possible.

### 8.3.2 Modbus TCP

Le protocole Modbus TCP se présente exactement de la même façon que le protocole Modbus RTU, à l'exception du code de contrôle des trames et de l'interprétation de l'adresse.

De nouveau, les communications sont basées sur le principe maître-esclave et le rack VM600 est l'esclave dans le système. L'équipement maître demande des données à l'esclave défini qui ne peut que répondre. L'esclave ne peut pas initier une transaction. Le maître peut gérer des esclaves individuels pour demander ou envoyer des données. Il peut également demander à ce qu'une mesure soit prise par un ou tous les esclaves du réseau.

## 8.4 Description du protocole Modbus RTU

### 8.4.1 Trame et horloge

**NOTE :** Les définitions suivantes du protocole Modbus sont adaptées au MPS du VM600.

Le protocole Modbus RTU est un format de transmission des données en série largement utilisé dans les communications avec les automates programmables. Il s'adapte facilement à d'autres types d'unités distantes grâce à la structure particulière de ses messages (c'est-à-dire qu'il ne fonctionne pas avec les variables, mais avec les adresses mémoire).

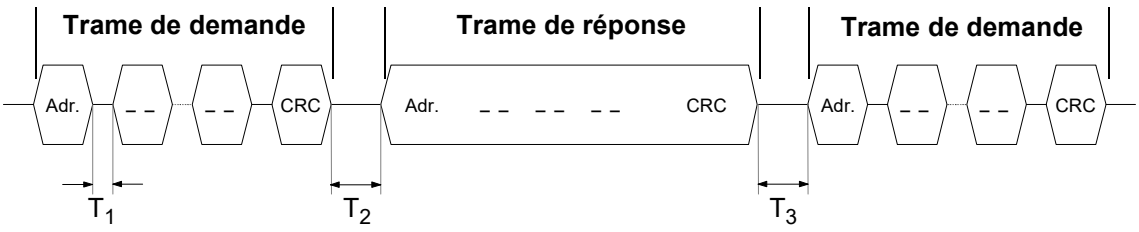
Comme indiqué dans la Figure 8-2, le premier caractère d'une trame est l'adresse esclave, suivie du code de fonction Modbus et du champ d'information. Finalement, deux octets sont réservés pour un code de contrôle des erreurs basé sur un contrôle par redondance cyclique (CRC).

Chaque caractère est composé de 10 bits : 1 bit de départ, 8 bits de données et 1 bit de stop.

**Figure 8-2:** Constitution d'une trame

Le protocole Modbus RTU n'utilise aucun caractère de délimitation au début ou à la fin d'un message. Chaque trame doit être précédée et suivie d'un intervalle silencieux d'au moins 3,5 fois le temps de transmission d'un caractère (character time, CT). Voir la Figure 8-3.

L'équipement connecté (rack VM600) détecte le début d'un message lorsqu'un caractère valide (contenant son adresse ou l'adresse 00) a été reçu après un intervalle silencieux d'au moins 3,5 fois le temps de transmission d'un caractère. La fin du message est interprétée par un intervalle silencieux de la même durée.




$T_1$ = délai entre deux caractères (min. = 0, max. = 3,5 CT)
$T_2$ = délai entre la demande et la réponse (min. 3,5 CT)
$T_3$ = délai entre la réponse et la prochaine demande (min. 3,5 CT)
CT = 1 fois le temps de transmission d'un caractère (= le temps requis pour transmettre un caractère composé de 10 bits)

| Bauds (bit/s) | Délai pour 3,5 CT |
|---|---|
| 1200 | 30 ms |
| 2400 | 15 ms |
| 4800 | 8 ms |
| 9600 | 4 ms |
| 19200 | 2 ms |

**Figure 8-3:** Considérations de temps

### 8.4.2 Paramètres de communication du VM600

Par défaut, les paramètres de communication du VM600 sont :

- Débit en bauds 19200
- Parité Even (Impaire)
- Nombre de bits 8
- Nombre de bits d'arrêt 1
- Adresse esclave 1.

Ces paramètres peuvent si nécessaire être modifiés en éditant le fichier de configuration approprié enregistré sur la CPUM (par exemple, « mbcfg.2 », « mbcfg.3 », « mbcfg.4 » ou « mbcfg.5 »). Pour ce faire, utilisez un terminal VT100 ou un émulateur de terminal de l'environnement Windows. Voir la section 2.3 Fichiers de configuration Modbus pour de plus amples informations.

### 8.4.3 Fonctions Modbus prises en charge par le VM600

Les fonctions Modbus prises en charge par le VM600 sont affichées dans le tableau Tableau 8-2.

**Tableau 8-2 :** Fonctions Modbus prises en charge

| Code de fonction Modbus | Description de la fonction |
|---|---|
| 01 (0x01) | Lecture des bits internes |
| 02 (0x02) | Lecture des entrées discrètes |
| 03 (0x03) | Lecture du registre de maintien |
| 04 (0x04) | Lecture des registres d'entrée |

Les applications de ces fonctions Modbus à un VM600 sont affichées dans le Tableau 8-3.

**Tableau 8-3 :** L'application des fonctions Modbus prises en charge par un VM600

| Code de fonction Modbus | Utilisation de la fonction |
|---|---|
| 01 | Utilisée pour lire les informations de statut ou de configuration (bit unique) du VM600, comme l'état alarme, danger et OK |
| 02 | Identique au code de fonction Modbus 01 |
| 03 | Utilisée pour lire les valeurs dynamiques et les informations de configuration (au format d'un entier de 16 bits) du VM600 |
| 04 | Identique au code de fonction Modbus 03 |

Lorsqu'elles sont utilisées pour lire les valeurs d'une carte MPC4, les fonctions Modbus 01 et 02 sont interchangeables et renvoient toutes deux le même résultat.

De même, lorsqu'elles sont utilisées pour lire les valeurs d'une carte MPC4, les fonctions Modbus 03 et 04 sont interchangeables. Elles renvoient toutes deux le même résultat.

**NOTE :** Si une fonction Modbus non gérée est demandée, un code d'exception « 01 » est renvoyé pour indiquer l'utilisation d'une fonction non autorisée.
Si une valeur située en dehors de la plage de configuration du VM600 est demandée, le code d'exception « 02 » est renvoyé pour indiquer l'utilisation d'une adresse non autorisée.

### 8.4.4 Formats de fonction Modbus

#### 8.4.4.1 Fonctions 01 et 02

Les bits internes et les entrées discrètes peuvent être lus à l'aide de la fonction 01 ou de la fonction 02 du protocole Modbus.

**NOTE :** Un maximum de 2040 valeurs discrètes (bits internes) peut être demandé en une fois.

Les trames de demande et de réponse sont affichées dans la Figure 8-4.

**Fonctions 01/02 (lecture des bits internes/lecture des entrées discrètes)**

| **Trame de demande** | | | | |
|---|---|---|---|---|
| 1 octet | 1 octet | 2 octets | 2 octets | 2 octets |
| Adresse esclave | Fonction Modbus | Première adresse de bit interne à être lue | Nombre de bits internes à être lus | CRC |

| **Trame de réponse** | | | | |
|---|---|---|---|---|
| 1 octet | 1 octet | 1 octet | n octets | 2 octets |
| Adresse | Fonction | Nombre d'octets | Valeurs de bit (le bit de poids faible est la première valeur de bit interne) | CRC |

**NOTE :** Lorsqu'une variable située en dehors de la plage de configuration du VM600 est demandée, le code d'exception « 02 » est renvoyé pour indiquer l'utilisation d'une adresse de donnée non autorisée.

Exemple:

Lecture de huit valeurs discrètes de l'adresse Modbus 01, à partir du premier bit interne (= bit interne 0).

Trame de demande

| **0x01** | **0x01** | **0x00** | **0x00** | **0x00** | **0x08** | **0xXX** | **0xXX** |
|---|---|---|---|---|---|---|---|
| Adresse esclave | Fonction Modbus | Première adresse de registre à être lue = 0x0000 | | 8 bits à être lus | | CRC | |

Un exemple de réponse si le bit interne a une valeur de 0b10010110 (binaire) comme suit :

0 est 0 Bit de poids faible (LSB)
1 est 1
2 est 1
3 est 0
4 est 1
5 est 0
6 est 0
7 est 1 Bit de poids fort (MSB)

Ceci correspond à une valeur de 0x96 (en hexadécimal).

Trame de réponse

| 0x01 | 0x01 | 0x01 | 0x96 | 2 octets |
|---|---|---|---|---|
| Adresse | Fonction | Nombre d'octets | 1 octet de données | CRC |

**Figure 8-4:** Formats et exemple pour la fonction Modbus 01

### 8.4.4.2 Fonctions 03 et 04

Les registres peuvent être lus à l'aide de la fonction 03 ou de la fonction 04 du protocole Modbus.

**NOTE :** Un maximum de 127 registres peut être demandé en une fois.

Les trames de demande et de réponse sont affichées dans la Figure 8-5.

**Fonction 03/04 (lecture de plusieurs registres)**

| **Trame de demande** | | | | |
|---|---|---|---|---|
| 1 octet | 1 octet | 2 octets | 2 octets | 2 octets |
| Adresse esclave | Fonction Modbus | Première adresse de registre à être lue | Nombre de mots (registres) à être lus | CRC |

| **Trame de réponse** | | | | |
|---|---|---|---|---|
| 1 octet | 1 octet | 1 octet | n octets | 2 octets |
| Adresse | Fonction | Nombre d'octets | Valeurs de registre | CRC |

**NOTE :** Lorsqu'une variable située en dehors de la plage de configuration du VM600 est demandée, le code d'exception « 02 » est renvoyé pour indiquer l'utilisation d'une adresse de donnée non autorisée.

Exemple:

Lecture d'un registre de l'adresse Modbus 01, à partir du registre 55 (en décimal).

Trame de demande

| 0x01 | 0x03 | 0x00 | 0x36 | 0x00 | 0x01 | 0xXX | 0xXX |
|---|---|---|---|---|---|---|---|
| Adresse esclave | Fonction Modbus | Première adresse de registre à être lue = 0x0036 (voir Note 1) | | 1 registre à être lu | | CRC | |

**Note 1 :**
Le registre 01 (en décimal) possède l'adresse de registre 0x00 (en hexadécimal)
Le registre 17 (en décimal) possède l'adresse de registre 0x10 (en hexadécimal)
Le registre 49 (en décimal) possède l'adresse de registre 0x30 (en hexadécimal) etc.

Voici un exemple de réponse lorsque le registre 55 possède une valeur de 4660 (en décimal), correspondant à 0x1234 en hexadécimal :

Trame de réponse

| 0x01 | 0x03 | 0x02 | 0x12 | 0x34 | 2 octets |
|---|---|---|---|---|---|
| Adresse | Fonction | Nombre d'octets | Valeur de registre | | CRC |

**Figure 8-5:** Formats et exemple pour la fonction Modbus 03

### 8.4.5 Traitement des erreurs

Il existe deux types de gestion des erreurs mis en œuvre par le serveur Modbus : les erreurs relatives aux communications et celles relatives au système.

#### 8.4.5.1 Erreurs de communication

Le serveur Modbus du VM600 fournit deux codes d'exception d'erreur (voir la Figure 8-6) :

- Si le client (maître) demande une fonction qui n'est pas gérée par le VM600, le code d'exception « 01 » est renvoyé. Ce code indique une exception d'erreur en réponse à une fonction non autorisée.
- Si le client (maître) demande une adresse qui n'est pas configurée dans le VM600, le code d'exception « 02 » est renvoyé. Ce code indique une exception d'erreur en réponse à une adresse de données non autorisée.

**NOTE :** Lorsqu'une erreur se produit, le serveur (esclave) renvoie en réponse un message comportant un code d'exception. Pour faire la distinction entre la demande et la réponse, l'octet le plus fort du code de fonction est paramétré sur « 1 » dans la réponse du message.

**NOTE :** Les tentatives de communication (d'accès) à une carte non connectée donneront lieu à l'erreur de communication suivante : des valeurs de données de 0 (zéro) seront renvoyées.

**Traitement des erreurs**

Exemple:

Lecture d'un registre de l'adresse Modbus 01, à partir du registre 256 (en décimal).

Trame de demande

| 0x01 | 0x03 | 0x00 | 0xFF | 0x00 | 0x01 | 0xXX | 0xXX |
|---|---|---|---|---|---|---|---|
| Adresse esclave | Fonction Modbus | Première adresse de registre à être lue = 0x00FF (voir Note 1) | | 1 registre à être lu | | CRC | |

**Note 1 :**
Le registre 1 (en décimal) possède l'adresse de registre 0x00
Le registre 17 (en décimal) possède l'adresse de registre 0x10
Le registre 256 (en décimal) possède l'adresse de registre 0xFF etc.

Voici un exemple de réponse si le registre 256 n'existe pas :

Trame de réponse

| 0x01 | 0x83 | 0x02 | 2 octets |
|---|---|---|---|
| Adresse | Fonction | Code d'exception | CRC |

**Figure 8-6:** Formats et exemple de message d'erreur

#### 8.4.5.2 Erreurs système

Toutes les autres erreurs du serveur Modbus du VM600, comme les erreurs de conversion, les capteurs non connectés ou les données non valides sont exprimées à l'aide de bits internes (bits d'état). Chaque carte et canal possède des bits détat. Plus important encore, Modbus se charge de mettre en correspondance ces bits et de les rendre disponibles.

Par exemple, le fait que les valeurs de vibration ne soient pas correctement acquises ou traitées n'entraîne aucune exception d'erreur Modbus. La validité des valeurs acquises et traitées doit être vérifiée à l'aide des bits d'état.

Les erreurs de configuration sont de deux types, les erreurs physiques et les erreurs logiques :

- Une erreur physique survient si le maître demande des données d'une carte manquante ou qui ne répond pas à la CPUM. Par exemple, cette erreur peut survenir si une carte ne répond pas en raison d'un défaut ou d'une reconfiguration matérielle.
- Une erreur logique survient si les données demandées pointent vers une carte ou une sortie incorrectement configurée.

Pour les deux types d'erreur de configuration, la réponse sera complètement renseignée par des zéros.

Pour les erreurs de configuration physique, le signal Erreur de configuration physique est activé.

Pour les erreurs de configuration logique, le signal Erreur de configuration logique est activé.

Chaque client Modbus dispose de ses propres signaux d'erreur. Le signal d'erreur sera désactivé à la première demande d'état qui n'est pas une demande d'état de configuration après la demande d'état d'erreur.

## 8.5 Description du protocole Modbus TCP

Le protocole Modbus TCP se présente exactement de la même façon que le protocole Modbus RTU, à l'exception du code de contrôle des trames (CRC) et de l'interprétation de l'adresse.

**NOTE :** Toutes les demandes sont envoyées à l'aide de TCP sur le port enregistré **502**.

Les demandes sont normalement envoyées en mode semi-duplex sur une connexion donnée. En fait, il n'y a aucun intérêt à envoyer des demandes supplémentaires sur une seule connexion alors qu'une réponse est en attente.

Le champ « adresse esclave » de Modbus est remplacé par un « identificateur d'unité » d'un seul octet. Celui-ci peut être utilisé pour communiquer à l'aide de périphériques comme des ponts et des passerelles qui utilisent une seule adresse IP pour prendre en charge plusieurs unités finales indépendantes. Le VM600 n'utilise pas cette fonctionnalité. Dans la réponse, ce champ est toujours paramétré sur la valeur reçue, mais il n'est pas vérifié par le système.

La demande et la réponse sont précédées de six octets affichés dans le Tableau 8-4.

**Tableau 8-4 :** Octets de préfixe de Modbus TCP

| Octets | Fonction |
|---|---|
| 0 | Identifiant de transaction (en général 0) copié par le serveur |
| 1 | Identifiant de transaction (en général 0) copié par le serveur |
| 2 | Identifiant de protocole = 0 |
| 3 | Identifiant de protocole = 0 |
| 4 | Champ de longueur (octet supérieur) = 0<br>(puisque tous les messages sont inférieurs à 256) |
| 5 | Champ de longueur (octet inférieur) = nombre d'octets qui suit |
| 6 | Identifiant d'unité (anciennement « adresse esclave ») – copié par le serveur mais ignoré |
| 7 | Code de fonction Modbus |
| À partir de 8 | Données selon la demande |

Exemple :

Une demande de lecture d'un registre à l'emplacement 55 (0x37 en hexadécimal), contenant une valeur de 4660 (0x1234 en hexadécimal), donne le résultat suivant :

| | | ← TCP → | | | | | | ← Modbus → | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Demande | : | 00 | 00 | 00 | 00 | 00 | 06 | 0F | 03 | 00 | 36 | 00 | 01 |
| Réponse | : | 00 | 00 | 00 | 00 | 00 | 05 | 0F | 03 | 02 | 12 | 34 | |
| *Octet* | *:* | *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* | *8* | *9* | *10* | *11* |

Voir la section 8.4 Description du protocole Modbus RTU pour obtenir des exemples de données.

Les utilisateurs familiarisés avec Modbus RTU doivent savoir que les zones de contrôle 'CRC-16' ou 'LRC' ne sont pas nécessaires dans Modbus TCP. Les mécanismes de contrôle de somme de TCP/IP et de la couche de liaison (par exemple Ethernet) sont utilisés pour vérifier le bon acheminement des paquets de données.

## 8.6 Carte MPC4

Les informations de cette section concernent l'implémentation de l'interface logicielle Modbus pour les racks VM600 utilisés par les cartes CPUM exécutant une version logicielle 067 ou antérieure.

La fonctionnalité améliorée Modbus est fournie par les cartes CPUM exécutant la version logicielle 071 ou ultérieure. Voir la section 9 Configuration d'une connexion Modbus (logiciel CPUM version 071 ou ultérieure) pour obtenir plus d'informations.

### 8.6.1 Définition des registres

L'allocation du registre est définie dans des fichiers enregistrés sur le CPUM qui sont lus au démarrage du rack VM600. Ces fichiers sont les suivants :

| | |
|---|---|
| `mbcfg.2` | Pour les communications Modbus RTU via le connecteur RS-232 (type D) à neuf broches sur le panneau avant de la carte CPUM. |
| `mbcfg.3` | Pour les communications Modbus RTU via le connecteur RS (RJ11) sur le panneau avant de la carte IOCN. |
| `mbcfg.4` | Pour les communications Modbus RTU via les connecteurs « A » (RJ11) sur le panneau avant de la carte IOCN. |
| `mbcfg.5` | Pour les communications Modbus RTU via les connecteurs « B » (RJ11) sur le panneau avant de la carte IOCN. |
| `mbcfg.tcp` | Pour les communications Modbus TCP via le connecteur Ethernet (RJ45) sur le panneau avant de la carte IOCN. |

Ces fichiers peuvent si nécessaire être modifiés en les éditant avec un terminal VT100 (ou un émulateur de l'environnement Windows). Voir la section 2.3 Fichiers de configuration Modbus pour de plus amples informations.

**NOTE :** Les fichiers `mbcfg.x` et le fichier `mbcfg.tcp2` ne doivent être édités que si vous avez besoin de modifier les paramètres de communication.

Chaque fichier est organisé en sept colonnes (voir la Figure 8-7):

| | |
|---|---|
| Slot | Il s'agit du numéro du logement occupé par l'unité d'acquisition de données (DAU), c'est-à-dire la carte, dans le rack VM600.<br>Il peut avoir une valeur comprise entre 3 et 14. |
| DAU | Indique le type de carte dans le logement :<br>MPC = une carte MPC4. |
| Output | Définit le canal et/ou la sortie de la carte concernée. Voir la section 8.6 Carte MPC4. |
| Type | Indique si la valeur (analogique) renvoyée est la valeur mesurée actuelle ou le paramètre d'échelle finale défini pour la sortie concernée :<br>• VAL = Valeur mesurée<br>• FSD = Valeur d'échelle finale |
| Sts/Not | Définit si les valeurs discrètes relatives à la sortie concernée sont demandées ou non :<br>• STS = Valeurs demandées (NB : STS = état)<br>• NOT = Valeurs non disponibles. |
| ANALOG. REG. NO. | Registre utilisé pour les valeurs analogiques relatives à la sortie concernée. |
| DIGITAL REG. NO. | Registre utilisé pour les valeurs discrètes (numériques) relatives à la sortie concernée. |

Les registres de valeurs analogiques et discrètes sont numérotés de manière séquentielle et en parallèle. Le fichier de configuration est scanné, la numérotation des deux séries de registres commençant par 1 (adresse = 0). À chaque fois qu'une déclaration de registre analogique ou une entrée discrète est rencontrée, le numéro de registre (numéro en cours) est incrément.

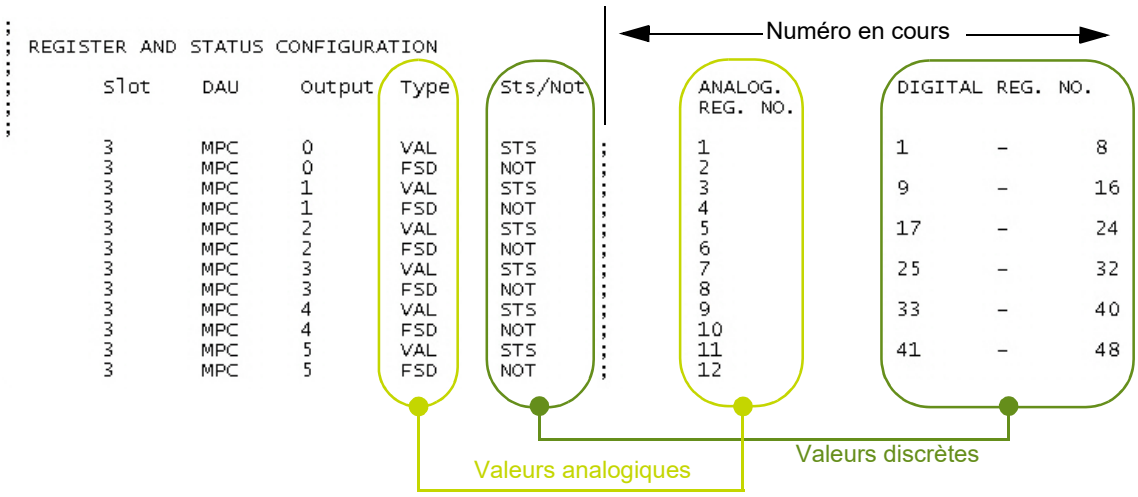



**Figure 8-7:** Extrait d'un fichier de configuration typique

### 8.6.2 Définition de sorties

Les sorties répertoriées dans la troisième colonne du fichier de configuration (voir la Figure 8-7) peuvent avoir onze valeurs possibles pour les cartes MPC4, comme défini dans le Tableau 8-5.

**Tableau 8-5 :** Codage du canal de sortie de la MPC4

| Numéro de sortie | Définition |
|---|---|
| 0 | Canal de mesure 1, sortie 1 |
| 1 | Canal de mesure 1, sortie 2 |
| 2 | Canal de mesure 2, sortie 1 |
| 3 | Canal de mesure 2, sortie 2 |
| 4 | Canal de mesure 3, sortie 1 |
| 5 | Canal de mesure 3, sortie 2 |
| 6 | Canal de mesure 4, sortie 1 |
| 7 | Canal de mesure 4, sortie 2 |
| 8 | Double canal 1 et 2, sortie 1 |
| 9 | Double canal 3 et 4, sortie 1 |
| 10 | Canal de vitesse 1, sortie 1 |
| 11 | Canal de vitesse 2, sortie 1 |

### 8.6.3 Valeurs analogiques (registres)

Les paramètres possibles sont :

- *VAL Valeur mesurée en cours*
  VAL est codé comme une valeur signée à 16 bits (voir la Figure 8-8) étalonnée comme suit :
  - 0 représente 0 % de la valeur de pleine échelle définie dans le registre FSD
  - 16383 (0x3FFF) représente 100% de la valeur de pleine échelle définie dans le registre FSD
  - 49152 (0xC000) représente –100% de la valeur de pleine échelle définie dans le registre FSD.

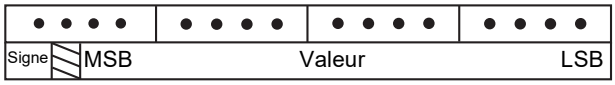



**Figure 8-8:** Entier signé à 16 bits

- *FSD Valeur de pleine échelle*
  - La valeur FSD est codée comme une valeur non signée à 16 bits (voir la Figure 8-9).

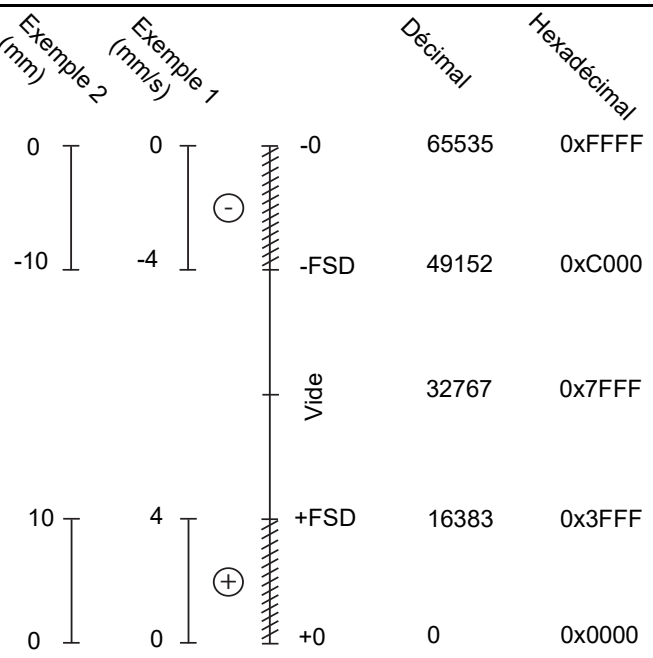



**Figure 8-9:** Valeurs analogiques

Exemple 1 :

Une valeur de 10000 est renvoyée avec une valeur de pleine échelle de 10.
(L'unité physique est exprimée en mm/s, mais ces informations ne sont pas disponibles dans Modbus.)

La valeur mécanique correspondante est ensuite calculée comme suit :

(10000 / 16383) x 10 = 6,10 mm/s.

Exemple 2 :

Une valeur de 50000 est renvoyée avec une valeur de pleine échelle de 20.
La valeur mécanique correspondante est ensuite calculée comme suit :

(50000 − 65535) / 16383 x 20 = −18,96 mm.

**NOTE :** Le deux valeurs de vitesse (sorties 10 et 11 du canal de vitesse dans le Tableau 8-5) sont codées différemment. La valeur étalonnée est la valeur RPM divisée par 4.
**Par conséquent, il vous suffit, pour obtenir des valeurs RPM, de multiplier la valeur renvoyée par 4**.

### 8.6.4 Valeurs discrètes (bits internes, entrées discrètes)

Les paramètres possibles sont :

- STS Valeurs d'état demandées
- NOT Valeurs d'état non disponibles.

La mise en correspondance de toutes les valeurs discrètes disponibles pour une sortie peut être activée ou non à l'aide de l'indication STS ou NOT.

Les huit valeurs discrètes disponibles pour la sortie seront mises en correspondance comme indiqué dans le Tableau 8-6.

**Tableau 8-6 :** Mise en correspondance des valeurs discrètes du registre de l'état du canal du MPC4

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b0 | Invalide | Le point n'est pas défini | Le point est défini |
| b1 | A+ | Le canal n'est pas en condition Alarme+ | Le canal est en condition Alarme+ |
| b2 | A– | Le canal n'est pas en condition Alarme– | Le canal est en condition Alarme– |
| b3 | D+ | Le canal n'est pas en condition Danger+ | Le canal est en condition Danger+ |
| b4 | D– | Le canal n'est pas en condition Danger– | Le canal est en condition Danger– |
| b5 | SOK | La vérification de la valeur OK du système a échoué | Capteur OK |
| b6 | | Réservé | |
| b7 | | Réservé | |

Les valeurs discrètes ne sont pas disponibles individuellement (par exemple, vous ne pouvez pas seulement demander le bit b3). Les huit bits doivent être demandés.

**NOTE :** L'implémentation du serveur Modbus pour les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) met en correspondance les bits d'état supplémentaires mis à disposition par la carte avec ce registre de l'état du canal (voir le Tableau 9-10).

### 8.6.5 Valeurs discrètes codées dans un registre de valeurs analogiques

Les valeurs discrètes présentées dans la 8.6.4 Valeurs discrètes (bits internes, entrées discrètes) sont également accessibles à l'aide des fonctions Modbus 03 et 04 pour les numéros de registre postérieurs à 1001 (voir le Tableau A-2 dans l'Annexe A : Définitions de registre Modbus MPC4). La mise en correspondance est identique à celle définie dans le Tableau 8-6 mais avec les bits b8 à b15 paramétrés sur zéro.

### 8.6.6 Mise en correspondance des adresses

Les définitions de registre de la carte MPC4 de Modbus sont disponibles dans les tableaux de mise en correspondance d'adresses présents dans l'Annexe A : Définitions de registre Modbus MPC4. Les adresses de démarrage du registre Modbus peuvent être définies à l'aide des informations fournies dans les tableaux.

## 8.7 Carte AMC8

Les informations de cette section concernent l'implémentation de l'interface logicielle Modbus pour les racks VM600 utilisés par les cartes CPUM exécutant une version logicielle 067 ou antérieure.

La fonctionnalité Modbus améliorée est fournie par les cartes CPUM exécutant la version logicielle 071 ou ultérieure. Voir la section 9 Configuration d'une connexion Modbus (logiciel CPUM version 071 ou ultérieure) pour obtenir plus d'informations.

Les sections suivantes décrivent les valeurs ou paramètres de configuration, soit discrets ou analogiques, pouvant être extraits à l'aide des quatre fonctions Modbus. (Voir la section 8.4.3 Fonctions Modbus prises en charge par le VM600 pour obtenir une liste de ces fonctions.)

### 8.7.1 Valeurs discrètes (bits internes)

La fonction Modbus 01 est la fonction lecture des registres de bits internes. Elle est utilisée pour lire la configuration des alarmes (valeurs discrètes) et vérifier si elles sont activées ou non. Ces configurations ont été effectuées dans le logiciel MPS sur l'écran **Traitements et alarmes** (pour les canaux) et sous l'écran **Alarmes** (pour les multi-canaux).

Les quatre sorties d'alarme pouvant être activées sont les suivantes :

- Alerte– Basse
- Alerte+ Haute
- Danger– Bas
- Danger+ Haut.

### 8.7.2 Valeurs discrètes (entrées discrètes)

La fonction Modbus 02 est la fonction Lecture des entrées discrètes. Elle est utilisée pour extraire les valeurs discrètes comme l'état des canaux (erreur et échec) et le registre des alarmes.

Les cinq registres suivants définissent si le système est en condition Alerte pour chaque canal et multi-canal ou s'il y a un échec global du OK du canal.

- Alarme+
- Alarme–
- Danger+
- Danger–
- Échec global du OK du canal.

Les dix registres suivants indiquent s'il y a une erreur ou un état de pause pour chaque canal et multi-canal.

- Erreur ADC
- Pause ADC
- Erreur de verrouillage PLL ADC
- Erreur de transmission ADC
- Erreur de configuration dynamique ADC
- Échec du résultat BIT
- Pas d'erreur d'échantillonnage
- Échec erreur OK
- Erreur de linéarisation
- Erreur de soudure à froid.

Le deux derniers registres définissent si chacune des 16 fonctions de base et chacune des huit fonctions avancées ont été activées sur l'écran **Combinaisons logiques d'alarmes** du logiciel MPS.

- Fonctions de base
- Fonctions avancées.

### 8.7.3 Registres de maintien

La fonction Modbus 03 est la fonction lecture des registres de maintien. Elle est utilisée pour lire les valeurs analogiques. Elle est en particulier utilisée pour lire les valeurs en cours de chaque canal de l'AMC8 et les résultats des combinaisons logiques d'alarmes.

Cette fonction, ainsi que la fonction 02, peuvent être utilisées pour lire l'état d'alarme.

La signification des registres d'entrée est définie dans le Tableau 8-7.

**Tableau 8-7 :** Lire les définitions du registre d'entrée

| Registre | Définition | |
|---|---|---|
| Valeurs traitées | La valeur en cours de chaque canal et multi-canal. | |
| Résultats logiques | Les résultats des combinaisons logiques définies sur l'écran **Combinaisons logiques d'alarmes** du MPS pour les fonctions de base de 1 à 16 et les fonctions avancées de 1 à 8 sont les suivants : | |
| | **Bit** | **Définition** |
| | 0 | BF1 |
| | … | … |
| | 15 | BF15 |
| | 16 | AF1 |
| | … | … |
| | 23 | AF8 |
| | 24 à 31 | 0 (non utilisé) |
| État alarme | Définit les alarmes et l'état du système à l'aide des définitions de bit suivantes : | |
| | **Bit** | **Définition** |
| | 0 | Alarme+ |
| | 1 | Alarme– |
| | 2 | Danger+ |
| | 3 | Danger– |
| | 4 | Échec global du OK du canal |
| | 5 | Erreur ADC |
| | 6 | Pause ADC |
| | 7 | Erreur de verrouillage PLL ADC |
| | 8 | Erreur de transmission ADC |
| | 9 | Erreur de configuration dynamique ADC |
| | 10 | Échec du résultat BIT |
| | 11 | Pas d'échantillonnage |
| | 12 | Échec erreur OK |
| | 13 | Erreur de linéarisation |
| | 14 | Erreur de soudure à froid |
| | 15 | Réservé |

### 8.7.4 Registres d'entrées

La fonction Modbus 04 est la fonction lecture des registres d'entrées. Elle est utilisée pour lire les valeurs de configuration comme les valeurs minimum et maximum affichées et les niveaux d'alarme pour chaque canal et multi-canal utilisant l'unité de sortie définie.

La liste complète des registres de maintien est affichée ci-dessous :

- Valeur minimum affichée
- Valeur maximum affichée
- Alerte–Niveau Bas
- Alerte+ Niveau Haut
- Danger– Niveau Bas
- Danger+ Niveau Haut
- Unité de sortie
  - Bit 0 : défini par l'utilisateur
  - Bit 1 : degrés Kelvin
  - Bit 2 : degrés Celsius
  - Bit 3 : degrés Fahrenheit
  - Bits 4 à 15 : non utilisés.

### 8.7.5 Mise en correspondance des adresses

Les définitions de registre de la carte AMC8 de Modbus sont disponibles dans les tableaux de mise en correspondance d'adresses présents dans l'Annexe B : Définitions de registre Modbus AMC8. Les adresses de démarrage de registre Modbus peuvent être définies à l'aide des informations fournies dans l'annexe.

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# 9 CONFIGURATION D'UNE CONNEXION MODBUS (LOGICIEL CPUM VERSION 071 OU ULTÉRIEURE)

## 9.1 Introduction

Ce chapitre décrit l'implémentation de l'interface logicielle Modbus pour les racks VM600 tel qu'utilisée par les cartes CPUM exécutant la version logicielle 071 ou ultérieure. C'est la dernière implémentation de l'interface du logiciel Modbus pour le VM600. Elle comprend des registres supplémentaires pour permettre la lecture d'un plus grand nombre d'informations à l'aide de Modbus.

Tous les registres Modbus sont disponibles pour les cartes CPUM exécutant la version logicielle 071 (ou ultérieure). Elle est également compatible avec les générations précédentes du serveur Modbus existant pour les cartes CPUM exécutant la version logicielle 067 (ou antérieure).

**NOTE :** Même si vous utilisez la dernière interface logicielle Modbus (pour les cartes CPUM exécutant la version logicielle 071 (ou ultérieure), vous devez également lire l'ensemble du 8 Configuration d'une connexion Modbus (logiciel CPUM version 067 ou antérieure).
Les informations fournies dans ce chapitre s'ajoutent aux informations déjà données pour les cartes CPUM exécutant la version logicielle 067 (ou antérieure).

En pratique, le serveur Modbus amélioré est bien plus flexible. Des extensions ont été ajoutées aux registres Modbus de façon à ce que les données supplémentaires puissent être fournies par le système VM600. Ces données supplémentaires comprennent :

- Les fonctions de base et avancées de la carte MPC4
- Les informations d'état de la carte MPC4
- Les informations d'état de la carte AMC8.

Cependant, les codes de fonction Modbus 01 (lecture de l'état des bits internes) et 02 (lecture de l'état des entrées) utilisés pour la lecture des entrées discrètes (individuelles d'un bit) n'ont pas changé.

Voir l'Annexe A : Définitions de registre Modbus MPC4 pour des informations détaillées sur le mappage du registre Modbus de la carte MPC4 et à l'Annexe B : Définitions de registre Modbus AMC8 pour des informations détaillées sur le mappage du registre Modbus de la carte AMC8.

### 9.1.1 Fonctions Modbus prises en charge par le VM600 – version 071 ou ultérieure

Outre les fonctions Modbus existantes (voir les 8.4.3 Fonctions Modbus prises en charge par le VM600), les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) fournissent les fonctions Modbus supplémentaires affichées dans le Tableau 9-1.

**Tableau 9-1 :** Fonctions Modbus supplémentaires prises en charge

| Code de fonction Modbus | Description de la fonction |
|---|---|
| 05 (0x05) | Écriture d'un bit interne |
| 06 (0x06) | Écriture d'un registre |
| 15 (0x0F) | Écriture de plusieurs bits internes |
| 16 (0x10) | Écriture de plusieurs registres |

**NOTE :** Si une fonction Modbus non gérée est demandée, un code d'exception « 01 » est renvoyé pour indiquer l'utilisation d'une fonction non autorisée.
Si une valeur située en dehors de la plage de configuration du VM600 est demandée, le code d'exception « 02 » est renvoyé pour indiquer l'utilisation d'une adresse non autorisée.

## 9.2 Définition des registres

La configuration a considérablement changé pour la version logicielle 071 (ou ultérieure). Les cinq fichiers de configuration distincts qui étaient auparavant utilisés ont été remplacés par un seul fichier de configuration.

`modbusDefault.cfg` Pour toutes les communications et configurations Modbus.

Comme précédemment, le fichier de configuration est un fichier de texte qui peut être lu ou modifié à l'aide de tout type d'éditeur de texte. Ce fichier peut par exemple être modifié en l'éditant avec un terminal VT100 (ou un émulateur de l'environnement Windows). Voir la section 2.3 Fichiers de configuration Modbus pour obtenir des informations sur l'utilisation de l'éditeur vi pour modifier les fichiers.

Cependant, du fait de la taille du fichier de configuration, la seule façon pratique d'effectuer des modifications est d'utiliser l'éditeur vi. Pour les changements plus significatifs, il est conseillé de modifier le fichier modbusDefault.cfg à l'aide d'un programme d'édition plus avancé sur un ordinateur. Le fichier de configuration édité peut ensuite être envoyé à la CPUM à l'aide d'un utilitaire logiciel appelé ModbusConfigSender.exe (développé et gratuitement mis à disposition par Meggitt).

**NOTE :** En général, le fichier `modbusDefault.cfg` n'est nécessaire que pour éditer les paramètres de communication.

Le contenu du fichier de configuration ne fait pas la différence entre majuscules et minuscules et est organisé selon les sections suivantes : [RTUx], [TCP], [GLOBAL] et [MAPPING].

### 9.2.1 La section RTUx

La section RTUx peut avoir une ou plusieurs sections appelées [RTU1], [RTU2], [RTU3].

Chaque section [RTUx] sera associée aux paramètres affichés dans le Tableau 9-2.

**Tableau 9-2 :** Paramètres de la section RTU

| Paramètre de la section RTUx | Valeur par défaut | Valeurs autorisées |
|---|---|---|
| ENABLE | N | Y/N (O/N) |
| ADDRESS | 1 | 1 à 247 |
| DEVICE | RS | A/B/RS<br>Tous les ports sont situés sur le IOC-N à l'arrière du rack VM600. |
| SPEED | 19200 | 1200 à 11500 |
| PARITY | Even | Even, None, Odd (Impaire, paire ou aucune) |
| STOP | 1 | 1 ou 2 |
| DATABITS | 8 | 7 ou 8 |

### 9.2.2 La section TCP

La section TCP sera associée aux paramètres affichés dans le Tableau 9-3.

**Tableau 9-3 :** Paramètres de la section TCP

| Paramètre de la section TCP | Valeur par défaut | Valeurs autorisées |
|---|---|---|
| ENABLE | N | Y/N (O/N) |
| PORT | 502 | 0 à 65535 |

### 9.2.3 La section GLOBAL

Même si les valeurs de 32 bits ne faisaient pas partie de la spécification Modbus d'origine, des extensions ont été mises en œuvre qui permettent d'extraire des valeurs de 32 bits à l'aide de Modbus et de deux registres consécutifs. L'ordre des octets des données contenues dans les deux registres consécutifs n'a cependant pas été normalisé de sorte que l'utilisateur doit indiquer l'ordre de son choix. L'utilisateur peut choisir d'extraire des valeurs à virgule flottante de 32 bits et des valeurs de long mot de 32 bits et ce dans différents formats.

**NOTE :** La section GLOBAL est facultative.

Les valeurs à virgule flottante de 32 bits peuvent être associées aux paramètres affichés dans le Tableau 9-4. Si le paramétrage de la section Global est manquant, les valeurs à virgule flottante seront par défaut associées à l'ordre FM1 (le plus souvent rencontré).

**Tableau 9-4 :** Paramètres de la section GLOBAL

| Paramètre de la section GLOBAL | Valeur par défaut | Valeurs autorisées |
|---|---|---|
| DEFAULT_FLOAT_ORDER | FM1 | FB/FL/FM1/FM2 |
| où<br>FB : Valeurs à virgule flottante (32 bits) classées au format « big endian ». Par exemple, ABCD.<br>FL : Valeurs à virgule flottante (32 bits) classées au format « little endian ». Par exemple, DCBA.<br>FM1 : Valeurs à virgule flottante (32 bits) classées au format « mixed endian 1 ». Par exemple, CDBA.<br>FM2 : Valeurs à virgule flottante (32 bits) classées au format « mixed endian 2 ». Par exemple, BADC. | | |

Les valeurs de long mot de 32 bits peuvent être associées aux paramètres affichés dans le Tableau 9-5. Si le paramétrage de cette section est manquant, les valeurs de long mot seront par défaut associées à l'ordre LB.

**Tableau 9-5 :** Paramètres de la section GLOBAL

| Paramètre de la section GLOBAL | Valeur par défaut | Valeurs autorisées |
|---|---|---|
| DEFAULT_LONG_ORDER | LB | LB/LL/LM1/LM2 |
| où<br>LB : Valeurs de long mot (32 bits) classées au format « big endian ». Par exemple, ABCD.<br>LL : Valeurs de long mot (32 bits) classées au format « little endian ». Par exemple, DCBA.<br>LM1 : Valeurs de long mot (32 bits) classées au format « mixed endian 1 ». Par exemple, CDBA.<br>LM2 : Valeurs de long mot (32 bits) classées au format « mixed endian 2 ». Par exemple, BADC. | | |

### 9.2.4 La section MAPPING

La section MAPPING possède la syntaxe suivante :

**Address:Function[:DataType[:Min:Max]] = SlotNr:CardType:ValueName**

**NOTE :** La section Min:Max est facultative et ne s'utilise que pour les valeurs étalonnées.

La configuration possible de chaque paramètre est affichée dans le Tableau 9-6.

**Tableau 9-6 :** Paramètres de la section MAPPING

| Paramètre de la section MAPPING | Valeurs autorisées | |
|---|---|---|
| Adresse | 0 à 65535 | |
| Fonction | Fonctions Modbus 01 à 06, 15 (0x0F) et 16 (0x10) | |
| DataType | L'une des valeurs suivantes : | |
| | B | Bit (1 bit) |
| | F | Flottante (32 bits). FM1 est le type d'ordre par défaut (CDBA). Voir la note 1. |
| | FB | Flottante (32 bits). Ordre au format « big endian » (ABCD) |
| | FL | Flottante (32 bits). Ordre au format « little endian » (DCBA) |
| | FM1 | Flottante (32 bits). Ordre au format « mixed endian 1 » (CDBA) |
| | FM2 | Flottante (32 bits). Ordre au format « mixed endian 2 » (BADC) |
| | L | Long mot (32 bits). LB est le type d'ordre par défaut (ABCD). Voir la note 2. |
| | LB | Long mot (32 bits). Ordre au format « big endian » (ABCD) |
| | LL | Long mot (32 bits). Ordre au format « little endian » (DCBA) |
| | LM1 | Long mot (32 bits). Ordre au format « mixed endian 1 » (CDBA) |
| | LM2 | Long mot (32 bits). Ordre au format « mixed endian 2 » (BADC) |
| | U | 16 bits non signés (0x0000 à 0xFFFF) |
| Min:Max | La section Min:Max est facultative et ne doit être utilisée que pour les valeurs étalonnées. Voir la note 3. | |
| SlotNr | Ceci dépend du rack. Par exemple, sur le système VM600 actuel doté du rack ABE 040/42, les logements de la carte de mesure sont numérotés de 3 à 14. | |
| CardType | L'une des valeurs suivantes : | |
| | AMC8 | Pour la carte de contrôle analogique à huit canaux |
| | MPC4 | Pour la carte de protection des machines à quatre canaux |
| | CMC16 | Pour la carte de contrôle de l'état à 16 canaux |
| | XMC16 | Pour la carte de contrôle étendu à 16 canaux |

**Notes**

**1.** Le DataType F peut être paramétré sur FB, FL, FM1 ou FM2. Voir la section 9.2.3 La section GLOBAL.

**2.** Le DataType L peut être paramétré sur LB, LL, LM1 ou LM2. Voir la section 9.2.3 La section GLOBAL.

**3.** Pour chaque valeur étalonnée, les valeurs minimum et maximum seront lisibles par Modbus au format virgule flottante. Voir la section 9.4.1 Valeurs analogiques (registres).

Un extrait du fichier `modbusDefault.cfg`, présentant l'organisation de certaines données est affiché dans la Figure 9-1.

```
[MAPPING]

////////////////////////////////////////////////////
////////////////// Slot3 ///////////////////////////
////////////////////////////////////////////////////

///////////////// Alarm and Status ///////////////

//  Slot3, Channel 1, Output 1
$MPC4S3C101Used = Slot3:MPC4:Config:C1:O1:Used
$MPC4S3C101Apos = Slot3:MPC4:C1:O1:A+
$MPC4S3C101Aneg = Slot3:MPC4:C1:O1:A-
$MPC4S3C101Dpos = Slot3:MPC4:C1:O1:D+
$MPC4S3C101Dneg = Slot3:MPC4:C1:O1:D-
$MPC4S3C101SOK  = Slot3:MPC4:Status:C1:SOK
$MPC4S3C101CME  = Slot3:MPC4:Status:C1:CME
$MPC4S3C101ISE  = Slot3:MPC4:Status:C1:ISE
$MPC4S3C101PSE  = Slot3:MPC4:Status:C1:PSE
$MPC4S3C101DSE  = Slot3:MPC4:Status:C1:DSE
$MPC4S3C101TRL  = Slot3:MPC4:Status:C1:TRL
$MPC4S3C101TOR  = Slot3:MPC4:Status:C1:TOR
$MPC4S3C101UsedAndApos = AND($MPC4S3C101Used, $MPC4S3C101Apos)
$MPC4S3C101UsedAndAneg = AND($MPC4S3C101Used, $MPC4S3C101Aneg)
$MPC4S3C101UsedAndDpos = AND($MPC4S3C101Used, $MPC4S3C101Dpos)
$MPC4S3C101UsedAndDneg = AND($MPC4S3C101Used, $MPC4S3C101Dneg)
$MPC4S3C101UsedAndSOK  = AND($MPC4S3C101Used, $MPC4S3C101SOK)
$MPC4S3C101UsedAndCME  = AND($MPC4S3C101Used, $MPC4S3C101CME)
$MPC4S3C101UsedAndISE  = AND($MPC4S3C101Used, $MPC4S3C101ISE)
$MPC4S3C101UsedAndPSE  = AND($MPC4S3C101Used, $MPC4S3C101PSE)
$MPC4S3C101UsedAndDSE  = AND($MPC4S3C101Used, $MPC4S3C101DSE)
$MPC4S3C101UsedAndTRL  = AND($MPC4S3C101Used, $MPC4S3C101TRL)
$MPC4S3C101UsedAndTOR  = AND($MPC4S3C101Used, $MPC4S3C101TOR)

00000:1:B = Slot3:MPC4:Config:C1:O1:Used
00001:1:B = $MPC4S3C101UsedAndApos
00002:1:B = $MPC4S3C101UsedAndAneg
00003:1:B = $MPC4S3C101UsedAndDpos
00004:1:B = $MPC4S3C101UsedAndDneg
00005:1:B = $MPC4S3C101UsedAndSOK
00006:1:B = 0                                          //because not define
00007:1:B = 0                                          //because not define

00000:2:B = Slot3:MPC4:Config:C1:O1:Used
00001:2:B = $MPC4S3C101UsedAndApos
00002:2:B = $MPC4S3C101UsedAndAneg
00003:2:B = $MPC4S3C101UsedAndDpos
00004:2:B = $MPC4S3C101UsedAndDneg
00005:2:B = $MPC4S3C101UsedAndSOK
00006:2:B = 0                                          //because not define
00007:2:B = 0                                          //because not define

//  Slot3, Channel 1, Output 2
$MPC4S3C102Used = Slot3:MPC4:Config:C1:O2:Used
$MPC4S3C102Apos = Slot3:MPC4:C1:O2:A+
```

**Figure 9-1:** Extrait d'un fichier de configuration typique `modbusDefault.cfg` présentant la syntaxe de mise en correspondance.

## 9.3 Traitement des erreurs

En plus du traitement déjà décrit des erreurs au niveau du serveur Modbus (voir la section 8.4.5 Traitement des erreurs), le comportement des cartes CPUM exécutant la version logicielle 071 (ou ultérieure) a été amélioré pour permettre de documenter les erreurs de communication supplémentaires.

### 9.3.1 Erreurs de communication

La dernière implémentation du serveur Modbus du VM600 fournit un code d'exception d'erreur supplémentaire :

- Si le client (maître) demande des informations auprès d'une carte VM600 qui n'est pas connectée, le code d'exception « 06 » est renvoyé. Ce code indique généralement une exception d'erreur en réponse à un périphérique esclave occupé. (Le maître doit retransmettre le message ultérieurement, lorsque l'esclave est de nouveau disponible).

**NOTE :** Lorsqu'une erreur se produit, le serveur (esclave) renvoie en réponse un message comportant un code d'exception. Pour faire la distinction entre la demande et la réponse, l'octet le plus fort du code de fonction est paramétré sur « 1 » dans la réponse du message.

**NOTE :** Les tentatives de communication (d'accès) à une carte non connectée donneront lieu à l'erreur de communication suivante : le code d'exception « 06 » sera renvoyé.

## 9.4 Carte MPC4

Outre les fonctionnalités existantes de la MPC4 (voir la section 8.4.3 Fonctions Modbus prises en charge par le VM600), les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) fournissent des fonctionnalités supplémentaires.

### 9.4.1 Valeurs analogiques (registres)

Il est possible d'étalonner des valeurs analogiques sur des valeurs non signées de 16 bits (le DataType U). Voir la section 8.6.3 Valeurs analogiques (registres) pour obtenir plus d'informations. L'étalonnage est paramétré à l'aide de la syntaxe Min:Max décrite dans la 9.2.4 La section MAPPING.

Min est la valeur minimum correspondant à 0x0000 dans le registre Modbus et Max la valeur maximum correspondant à 0xFFFF dans le registre Modbus. Les valeurs en dehors de la plage Min:Max sont si nécessaire rattachées à 0x0000 et 0xFFFF.

Les valeurs de sortie minimum et maximum de chaque canal sont totalement indépendantes des autres valeurs minimum et maximum utilisée par le système VM600.

#### 9.4.1.1 Lecture des valeurs minimum et maximum

Il est possible de lire les valeurs minimum et maximum configurées pour un canal utilisant Modbus :

**Address:Function=MIN:Address:Function**

ou

**Address:Function=MAX:Address:Function**

Cependant, elles peuvent seulement être lues sous forme de chiffres à virgule flottante (le DataType F) L'exemple suivant contribue à illustrer la procédure.

Exemple :

Les données de configuration de la section de mise en correspondance d'un fichier de configuration typique :

```
...
30003:4:U:-50.50:250.0=Slot10:AMC8:C4:V
30004:4:F=MAX:30003:4
30006:4:F=MIN:30003:4
...
```

La première ligne de ces données est analysée dans le Tableau 9-7, la seconde ligne dans le Tableau 9-8 et la troisième ligne dans le Tableau 9-9.

**Tableau 9-7 :** Exemple de lecture des valeurs minimum et maximum – première ligne

| Données | Syntaxe | Description |
|---|---|---|
| 30003 | Address | Adresse Modbus |
| 4 | Function | Fonction Modbus 04 |
| U | DataType | Type de données non signées à 16 bits |
| –50.50 | Min | FSD minimum |
| 250 | Max | FSD maximum |
| = | | Équivaut à |
| 10 | SlotNr | Numéro de logement |
| AMC8 | CardType | Carte de contrôle analogique à huit canaux |
| C4 | | Canal |
| V | | Valeur de donnée |

Comme le montre l'analyse dans le Tableau 9-7, cette ligne du fichier de configuration est utilisée pour configurer la valeur des données de vibration du canal 4 de la carte AMC8 située dans le logement 10 du rack.

Ce paramètre est disponible à l'adresse Modbus 30003 sous forme d'une valeur non signée de 16 bits à l'aide de la fonction Modbus 04.

Le paramètre est une valeur étalonnée avec une plage de valeurs réelles comprises entre –50,50 et 250,00. Cette plage de valeurs réelles correspond à la valeur non signée de 16 bits (voir la section 8.6.3 Valeurs analogiques (registres)) suivante :

La valeur FSD minimum de –50,50 est associée au code 0x0000 dans le registre Modbus

La valeur FSD maximum de 250 est associée au code 0xFFFF dans Modbus.

Ces valeurs correspondant aux valeurs minimum et maximum, les valeurs réelles inférieures à –50,50 seront rattachées à 0x0000 et les valeurs supérieures à 250 seront rattachées à 0xFFFF.

**NOTE :** Contrairement à la version logicielle 067 (ou antérieure), les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) prennent en charge des valeurs d'étalonnage non symétriques. En d'autres termes, les valeurs FSD minimum et maximum ne doivent plus nécessairement avoir la même grandeur.

**Tableau 9-8 :** Exemple de lecture des valeurs minimum et maximum – seconde ligne

| Données | Syntaxe | Description |
|---|---|---|
| 30004 | Address | Adresse Modbus |
| 4 | Function | Fonction Modbus |
| F | DataType | Type de données |
| = | | Équivaut à |
| MAX | | |
| 30003 | | |
| 4 | | |

Le Tableau 9-8 présente une lecture de la valeur d'étalonnage maximum du canal à l'aide de Modbus. La valeur est codée comme un type de donnée F (et n'est par conséquent disponible que sous forme dune valeur flottante) de sorte qu'elle nécessite deux adresses consécutives à 16 bits. Les valeurs sont disponibles à l'aide de la fonction Modbus 04 aux adresses 30004 et 30005.

**Tableau 9-9 :** Exemple de lecture des valeurs minimum et maximum – troisième ligne

| Données | Syntaxe | Description |
|---|---|---|
| 30006 | Address | Adresse Modbus |
| 4 | Function | Fonction Modbus |
| F | DataType | Type de données |
| = | | Équivaut à |
| MIN | | |
| 30003 | | |
| 4 | | |

Le Tableau 9-9 présente une lecture de la valeur d'étalonnage minimum du canal à l'aide de Modbus. La valeur est codée comme un type de donnée F (et n'est par conséquent disponible que sous forme dune valeur flottante) de sorte qu'elle nécessite deux adresses consécutives à 16 bits. Les valeurs sont disponibles à l'aide de la fonction Modbus 04 aux adresses 30006 et 30007.

#### 9.4.1.2 Valeurs constantes

Il est possible de paramétrer les adresses Modbus sur des valeurs constantes. Des constantes peuvent être définies pour les types de donnée suivants :

- B (valeur de 1 bit)
- F (valeur flottante de 32 bits)
- U (valeur non signée de 16 bits).

Exemples :

Les données de la section de mise en correspondance d'un fichier de configuration typique :

```
30004:1:B=1
```

Les valeurs valides du type de données B sont 0 ou 1.

```
30005:4:F=10.0
```

Les valeurs valides du type de données F peuvent correspondre à toute valeur à virgule flottante de 32 bits.

```
30004:4:U=1234
```

Les valeurs valides du type de données U sont comprises entre 0 (0x0000) et 65535 (0xFFFF).

#### 9.4.1.3 Variables de bits

L'utilisateur peut à présent définir des variables de bits.

##### 9.4.1.3.1 Noms de variables de bits

Les noms de variables de bits doivent se composer des lettres « a à Z », des chiffres « 0 à 9 » et du caractère de soulignement « _ ». Aucun autre caractère spécial n'est autorisé et les noms de variables de bits ne font pas la différence entre majuscules et minuscules. Les règles de nommage des variables de bits sont les suivantes :

Chaque nom de variable de bit doit comporter le préfixe « $ ».

Le nom de variable ne peut pas commencer par un chiffre.

La longueur maximum est de 40 caractères.

Exemples :

Noms de variables de bits valides :

```
$abcd102
$Vibration_channel_2
$Easy
$OK_SOK_VALUE234
```

Noms de variables de bits invalides :

```
$Channel1?!!!
$Channel 0
$3Channel
```

##### 9.4.1.3.2 Opérations avec les variables de bits

Différentes opérations peuvent être effectuées avec les variables de bits.

Attribution des valeurs de bits. Par exemple :

```
$Variable = SlotNr:CardType:ValueName
```

Opérations de bits AND, OR, NOT et XOR. Par exemple :

```
$a=AND($b,$c,$d)
$o=OR($o1,$o2,$o3,$o4,$o5,$o6,$o7,$o8,$o9,$o10)
$x=NOT($z)
```

Les règles relatives aux opérations de bits effectuées sur les variables définies par l'utilisateur sont les suivantes :

- Une ligne ne peut avoir qu'un seul opérateur.
- Entre 2 et 16 arguments sont autorisés pour les opérations AND, OR et XOR.
- Un seul argument est autorisé pour l'opération NOT.
- Une virgule doit être utilisée comme séparateur entre les variables.
- Une variable ou une liste de variables doit être entre parenthèses.
- Les équations seront évaluées de haut en bas.
- Les définitions récursives (par exemple, `$x=NOT($x)`) ne sont pas prises en charge.

La compression de bits en un entier non signé de 16 bits est une fonction spéciale. Par exemple :

```
Address:Function:U=PACK($a,$b,$c,$d,$e,$f,$g,$h,$i,$j,$k,$l $m, 0, 0, 1)
```

Comme affiché ci-dessus, les constantes (à savoir 0 ou 1) peuvent également être utilisées.

Les règles relatives à la compression des variables définies par l'utilisateur sont les suivantes :

- Le premier paramètre transmis est le MSB (`$a` dans l'exemple ci-dessus) et le dernier est le LSB (`$p` dans l'exemple ci-dessus).
- Les constantes 0 et 1 sont autorisées.
- Une virgule doit être utilisée comme séparateur entre les variables.
- Une liste de variables doit être entre parenthèses.
- Les fonctions valides sont 3 et 4.

La décompression de bits à partir d'un entier non signé de 16 bits est une fonction spéciale. Par exemple, procédez comme suit pour décompresser des bits à partir d'un entier non signé de 16 bits :

```
Address:Function:B=UNPACK(address, function, index)
$bitvariable=UNPACK(address, function, index)
```

Où :

- Address indique le registre du code de fonction spécifié.
- Function indique le code de fonction Modbus.

- L'index donne la position dans l'entier non signé de 16 bits et doit être compris entre 1 et 16. La valeur 1 correspond au LSB.
- Les fonctions Modbus valides d'accès à l'opération de décompression sont les fonctions 1 et 2.

Attribution de valeurs aux adresses Modbus : Par exemple :

```
Address:Function:B =$Value
```

Le type de données étant B, seules les valeurs de bits sont autorisées.

Attribution de constantes à une variable de bit. Par exemple :

```
$Variable = 0
```

Les valeurs valides sont 0 et 1.

### 9.4.2 Valeurs discrètes (bits internes, entrées discrètes)

Outre les valeurs discrètes Modbus existantes (voir 8-15), les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) fournissent des valeurs supplémentaires.

#### 9.4.2.1 Registre de l'état du canal

Les valeurs discrètes disponibles pour le registre de l'état du canal du MPC4 sont mises en correspondance comme indiqué dans le Tableau 9-10. (Ce tableau comprend les huit valeurs discrètes disponibles avec la version 067 ou antérieure. Voir le Tableau 8-6 dans la section 8.6.2 Définition de sorties.)

**Tableau 9-10 :** Mise en correspondance des valeurs discrètes du registre de l'état du canal du MPC4

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b0 | Invalid | Le point n'est pas défini | Le point est défini |
| b1 | A+ | Le canal n'est pas en condition Alarme+ | Le canal est en condition Alarme+ |
| b2 | A– | Le canal n'est pas en condition Alarme– | Le canal est en condition Alarme– |
| b3 | D+ | Le canal n'est pas en condition Danger+ | Le canal est en condition Danger+ |
| b4 | D– | Le canal n'est pas en condition Danger– | Le canal est en condition Danger– |

**Tableau 9-10 :** Mise en correspondance des valeurs discrètes du registre de l'état du canal du MPC4(Suite)

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b5 | SOK | La vérification du OK du système a échoué | Capteur OK |
| Des informations d'état supplémentaires ont été ajoutées à ce registre : | | | |
| b6 | CME | Aucune erreur de rejet du mode commun | Erreur de rejet du mode commun |
| b7 | ISE | Aucune erreur de saturation d'entrée | Erreur de saturation d'entrée |
| b8 | PSE | Aucune erreur de saturation PGA | Erreur de saturation PGA |
| b9 | DSE | Aucune erreur de saturation DSP | Erreur de saturation DSP |
| b10 | TRL | Aucune erreur de perte du suivi MPC | Erreur de perte du suivi MPC |
| b11 | TOR | Aucune erreur de suivi hors gamme | Erreur de suivi hors gamme |
| b12 à b15 | Non utilisé | | |

**NOTE :** Les bits b6 à b15 ne sont pas accessibles à l'aide des fonctions Modbus 01 ou 02.

Les valeurs discrètes ne sont pas disponibles individuellement. Par exemple, vous ne pouvez pas demander le bit b3 seulement. Les huit bits doivent être demandés.

### 9.4.2.2 Registre de l'état du canal de vitesse

Les valeurs discrètes disponibles pour le registre de l'état du canal de vitesse du MPC4 sont mises en correspondance comme indiqué dans le Tableau 9-11.

**Tableau 9-11 :** Mise en correspondance des valeurs discrètes du registre de l'état du canal de vitesse du MPC4

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b0 | Invalid | Le point n'est pas défini | Le point est défini |
| b1 | A+ | Le canal n'est pas en condition Alarme+ | Le canal est en condition Alarme+ |
| b2 | A– | Le canal n'est pas en condition Alarme– | Le canal est en condition Alarme– |
| b3 | Non utilisé | | |
| b4 | Non utilisé | | |
| b5 | SOK | La vérification du OK du système a échoué | Capteur OK |
| Un état supplémentaire a été ajouté à ce registre : | | | |
| b6 | SOL | Aucune erreur de vitesse hors limite | Erreur de vitesse hors limite |
| b7 | ERR | Aucune erreur | Erreur |
| b8 à b15 | Non utilisé | | |

**NOTE :** Les bits b6 à b15 ne sont pas accessibles à l'aide des fonctions Modbus 01 ou 02.

Les valeurs discrètes ne sont pas disponibles individuellement. Par exemple, vous ne pouvez pas demander le bit b3 seulement. Les huit bits doivent être demandés.

### 9.4.3 État de la carte

Les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) fournissent des informations d'état pour la carte MPC4. Les registres d'état en 16 bits ont été mis en correspondance pour accéder à l'état de la carte.

#### 9.4.3.1 Registre d'état 1 – fonctions de base et avancées

Les fonctions de base (BF) et les fonctions avancées (AF) disponibles sont comme indiqué dans le Tableau 9-12.

**Tableau 9-12 :** Mise en correspondance du registre 1 de l'état de la carte

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b0 | BF1 | Fonction de base 1 inactive | Fonction de base 1 active |
| b1 | BF2 | Fonction de base 2 inactive | Fonction de base 2 active |
| b2 | BF3 | Fonction de base 3 inactive | Fonction de base 3 active |
| b3 | BF4 | Fonction de base 4 inactive | Fonction de base 4 active |
| b4 | BF5 | Fonction de base 5 inactive | Fonction de base 5 active |
| b5 | BF6 | Fonction de base 6 inactive | Fonction de base 6 active |
| b6 | BF7 | Fonction de base 7 inactive | Fonction de base 7 active |
| b7 | BF8 | Fonction de base 8 inactive | Fonction de base 8 active |
| b8 | AF1 | Fonction avancée 1 inactive | Fonction avancée 1 active |
| b9 | AF2 | Fonction avancée 2 inactive | Fonction avancée 2 active |
| b10 | AF3 | Fonction avancée 3 inactive | Fonction avancée 3 active |
| b11 | AF4 | Fonction avancée 4 inactive | Fonction avancée 4 active |
| b12 à b15 | Non utilisé | | |

Adresse Modbus : 1144 + ( ( logement − 3 ) * 3 ) + 0.

Fonction Modbus : 03 ou 04.

Ces informations sont accessibles à l'aide du canal « Fonction de base et avancée » comme défini dans le Tableau A-2 (disponible aux adresses 1144, 1147, 1150, …, 1178).

### 9.4.3.2 Registre d'état 2 – diagnostic et état commun n°1

Comme indiqué dans le Tableau 9-13.

**Tableau 9-13 :** Mise en correspondance du registre 2 de l'état de la carte

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b0 (= LSB) | CMF | Aucun échec de contrôle commun | Échec de contrôle commun |
| b1 | COF | Aucun échec commun du OK du capteur | Échec commun du OK du capteur |
| b2 | CPE | Aucune erreur de traitement commune | Erreur de traitement commune |
| b3 | CIE | Aucune erreur de signal d'entrée commune | Erreur de signal d'entrée commune |

**Tableau 9-13 :** Mise en correspondance du registre 2 de l'état de la carte(Suite)

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b4 | CA | Aucune alarme commune | Alarme commune |
| b5 | CD | Aucun danger commun | Danger commun |
| b6 | CSOL | Aucune vitesse hors limite commune | Vitesse hors limite commune |
| b7 | CTL | Aucune perte de suivi commune | Perte de suivi commune |
| b8 | CTOR | Aucun suivi hors gamme commun | Suivi hors gamme commun |
| b9 | CDSE | Aucune erreur de saturation DSP commune | Erreur de saturation DSP commune |
| b10 | CISE | Aucune erreur de saturation d'entrée commune | Erreur de saturation d'entrée commune |
| b11 | CCME | Aucune erreur CMR commune | Erreur CMR commune |
| b12 | SL | Aucun état mémorisé | État mémorisé |
| b13 | DTM | Entrée rehaussement des alarmes inactive | Entrée rehaussement des alarmes active |
| b14 | DBP | Entrée neutralisation de danger inactive | Entrée neutralisation de danger active |
| b15 | AR | Entrée réinitialisation d'alarme inactive | Entrée réinitialisation d'alarme active |

Adresse Modbus : 1144 + ( ( logement − 3 ) * 3 ) + 1.

Fonction Modbus : 03 ou 04.

Ces informations sont accessibles à l'aide du canal « État de carte n°1 » comme défini dans le Tableau A-2 (disponible aux adresses 1145, 1148, 1151, …, 1179).

#### 9.4.3.3 Registre d'état 3 – diagnostic et état commun n°2

Comme indiqué dans le Tableau 9-14.

**Tableau 9-14 :** Mise en correspondance du registre 3 de l'état de la carte

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b0 | MCR | Configuration MPC hors fonction | Configuration MPC en fonctionnement |
| b1 à bit 15 | Non utilisé | | |

Adresse Modbus : 1144 + ( ( logement − 3 ) * 3 ) + 2.

Fonction Modbus : 03 ou 04.

Ces informations sont accessibles à l'aide du canal « État de carte n°2 » comme défini dans le Tableau A-2 (disponible aux adresses 1146, 1149, 1152, …, 1180).

### 9.4.4 Mise en correspondance des adresses

Les définitions de registre de la carte MPC4 de Modbus sont disponibles dans les tableaux de mise en correspondance d'adresses présents dans l'Annexe A : Définitions de registre Modbus MPC4. Les adresses de démarrage du registre Modbus peuvent être définies à l'aide des informations fournies dans l'annexe.

## 9.5 Carte AMC8

Outre les fonctionnalités existantes de l'AMC8 (voir la section 8.7 Carte AMC8), les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) fournissent des fonctionnalités supplémentaires.

### 9.5.1 Lecture des registres de maintien

Pour être cohérent avec le comportement des cartes MPC4, un registre de 16 bits a été ajouté pour chaque sortie de canal et multi-canal mettant en correspondance les informations définies dans le Tableau 9-15.

**Tableau 9-15 :** Mise en correspondance des registres de maintien de l'état du canal de AMC8

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b0 | A– | Le canal n'est pas en condition Alarme– | Le canal est en condition Alarme– |
| b1 | A+ | Le canal n'est pas en condition Alarme+ | Le canal est en condition Alarme+ |
| b2 | D– | Le canal n'est pas en condition Danger– | Le canal est en condition Danger– |
| b3 | D+ | Le canal n'est pas en condition Danger+ | Le canal est en condition Danger+ |
| b4 à b15 | Non utilisé | | |

Décalage du registre d'adresse (Aoff) : de 38 à 49.

Fonction Modbus : 03 ou 04.

Ces informations sont accessibles à l'aide des canaux de « Canal 1 » à « Canal 8 » comme défini dans le Tableau B-3 (disponible aux adresses 4903, 4904, …, 7730).

4903 est le premier canal du premier logement et 7730 le dernier canal du dernier logement.

### 9.5.2 État de la carte

Les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) fournissent des informations d'état supplémentaires pour la carte AMC8. Un seul registre d'état en 16 bits a été mis en correspondance pour accéder à l'état de la carte.

#### 9.5.2.1 Registre d'état 1 de la carte AMC8 – fonctions de base et avancées

Comme indiqué dans le Tableau 9-16.

**Tableau 9-16 :** Mise en correspondance du registre 1 de l'état de la carte pour la MPC4

| Bit | État | Signification en état de désactivation (0) | Signification en état d'activation (1) |
|---|---|---|---|
| b0 | ACNR | Configuration AMC en fonctionnement | Configuration AMC hors fonction |
| b1 | AR | Entrée réinitialisation d'alarme inactive | Entrée réinitialisation d'alarme active |
| b2 | DBP | Entrée neutralisation de danger inactive | Entrée neutralisation de danger active |
| b3 | SL | Aucun état mémorisé | État mémorisé |
| b4 à b15 | Non utilisé | | |

Décalage du registre d'adresse (Aoff) : 50

Fonction Modbus : 03 ou 04.

Ces informations sont accessibles à l'aide du canal « Fonction de base et avancée » comme défini dans le Tableau A-2 (disponible aux adresses 4915, 5171, 5427, …, 7731).

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# 10 CONFIGURATION D'UNE CONNEXION PROFINET (LOGICIEL CPUM VERSION 081 OU ULTÉRIEURE)

## 10.1 Introduction

Ce chapitre décrit l'utilisation de l'interface logicielle PROFINET pour les racks VM600 utilisés par les cartes CPUM exécutant la version 081 ou ultérieure du microprogramme. L'ensemble des registres et des données disponibles via Modbus peuvent être disponibles via PROFINET pour les cartes CPUM exécutant cette version du microprogramme.

Ce manuel décrit l'interface logicielle PROFINET qui prend en charge l'interface de communication Ethernet industrielle pour un rack VM600 contenant les cartes suivantes :

- MPC4
- AMC8.

Voir l'Annexe C : Définition des logements PROFINET MPC4 pour des informations détaillées sur le mappage du PROFIBUS de la carte MPC4 et à l'Annexe D : Définitions des logements PROFINET AMC8 pour des informations détaillées sur le mappage du PROFIBUS de la carte AMC8.

Le tableau de l'Annexe E : Noms des symboles inclut une liste de balises descriptives prédéfinies à utiliser avec un système VM600. Celles-ci peuvent être utilisées comme noms symboliques pour permettre d'identifier et de retrouver plus facilement le contenu d'un registre (voir la Figure 10-17).

Ce chapitre comprend un exemple qui explique comment inclure un rack VM600 dans un réseau PROFINET à l'aide du logiciel Siemens SIMATIC STEP 7 et d'un contrôleur SIMATIC S7 spécifique. Les principes indiqués peuvent s'appliquer à tous les systèmes PROFINET.

## 10.2 PROFINET

PROFINET est un standard pour l'Ethernet industriel défini par PROFINET International (PI), un organisme multiconstructeurs responsable des deux protocoles PROFIBUS et PROFINET.

PROFINET se base sur l'Ethernet industriel, utilisé depuis de nombreuses années dans les usines de production, et qui porte Ethernet jusqu'au niveau du terrain, permettant une interconnexion directe avec les équipements de terrain (comme les capteurs ou les actionneurs). Dans la plupart des cas, les composants d'infrastructure du réseau existants (commutateurs réseaux, câbles, connecteurs) sont compatibles et peuvent être utilisés avec PROFINET.

Nous vous conseillons de consulter le site Internet PI pour les dernières informations mises à jour sur PROFINET :
http://www.profibus.com

### 10.2.1 Protocoles PROFINET

PROFINET fait appel à un concept modulaire et fait la distinction entre différents types de communication afin de satisfaire les diverses exigences en termes de temps de réponse :

- TCP/IP pour PROFINET CBA (« Component Based Automation »)
  Pas d'exigences de performance spécifiques : durée de cycle de l'ordre de 100 ms.
  Utilisé typiquement pour des opérations de lecture/écriture de configuration et lecture/écriture acycliques (c'est-à-dire non programmées à l'avance et à la demande).

- RT (« Real-time ») pour PROFINET CBA et PROFINET IO
  Exigences de performance accrues : temps de cycle jusqu'à 10 ms.
  Utilisé typiquement pour des opérations standard de transfert de données et d'alarmes cycliques (c'est-à-dire programmées à l'avance et répétitives).
- IRT (« Isochronous Real-time ») PROFINET IO
  Exigences de performance maximales : temps de cycle inférieurs à 1 ms.
  Utilisé typiquement pour les applications de contrôle-commande en temps réel.

PROFINET fait de plus la distinction entre les différents types de communication en fonction des nœuds à connecter :

- PROFINET CBA
  Convient pour la communication entre automates via TCP/IP ainsi qu'avec des exigences supplémentaires RT et est destiné surtout aux systèmes d'automatisation distribués (haut niveau).
- PROFINET IO
  Supporte la communication entre automates et dispositifs d'entrées/sorties (« IO ») avec des exigences RT/IRT et est destiné surtout à l'échange des données entre automates programmables (bas niveau).

**NOTE :** Il est possible d'utiliser tous les types de communication en parallèle sur un même réseau, c'est-à-dire que PROFINET IO et PROFINET CBA peuvent communiquer au même moment et sur le même bus.

VM600 CPUM utilise PROFINET IO pour supporter les connexions PROFINET.

### 10.2.2 Implémentation de PROFINET sur le VM600

L'interface logicielle PROFINET implémentée pour les cartes CPUM exécutant la version 081 ou ultérieure du microprogramme s'appuie sur l'interface Modbus existante décrite dans la section 8.2 Modbus.

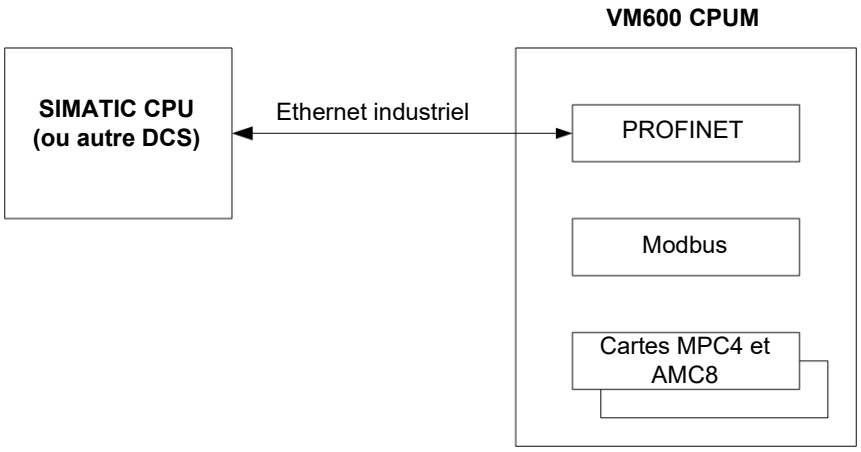



**Figure 10-1:** Implémentation de PROFINET sur la carte VM600 CPUM

Comme indiqué sur la Figure 10-1, la carte VM600 CPUM inclut une couche PROFINET supplémentaire au-dessus du serveur Modbus existant. En externe, cette couche PROFINET utilise le protocole d'E/S PROFINET pour gérer toutes les communications

PROFINET via une interface Ethernet industrielle. En interne, cette couche PROFINET utilise l'interface Modbus existante pour communiquer avec toutes les cartes du rack VM600.

Seule l'interface réseau principale (ETH1) d'une carte CPUM peut être utilisée pour les communications PROFINET. Voir la section 3.4.2.1 Exemple de fichier net.cfg pour plus d'informations.

**NOTE :** Pour que les communications PROFINET fonctionnent, seule l'interface réseau principale (ETH1) de la carte CPUM peut être utilisée.

En outre, pour plus d'informations sur l'activation et la configuration des communications PROFINET, voir la section 10.6 Fichiers de configuration PROFINET.

## 10.3 Communication PROFINET

À l'instar des autres standards Ethernet industriels, comme Modbus TCP et Ethernet/IP, PROFINET utilise TCP/IP et fait partie des normes IEC 61158 (Réseaux de communication industriels – Spécifications du bus de terrain) et IEC 61784 (Communication de données numériques à des fins de mesure et de contrôle).

Selon les performances exigées pour le système PROFINET, il est possible d'utiliser l'un des protocoles suivants :

- TCP/IP pour PROFINET CBA
- RT (« Real-time ») pour PROFINET CBA et PROFINET IO
- IRT (« Isochronous Real-time ») PROFINET IO.

La plupart des communications PROFINET utilisent des paquets TCP/IP non modifiés (Ethernet classique). Ceci permet une intégration aisée dans les réseaux existants, par exemple dans une usine ou un bureau.

Le protocole RT contourne TCP/IP et utilise une pile de protocoles optimisée pour donner la priorité aux paquets Ethernet. Ceci est utilisé typiquement pour accélérer les communications avec les automates programmables (CPU).

Le protocole IRT utilise une extension de la pile Ethernet pour synchroniser tous les équipements de communication. Ceci autorise une communication à haute vitesse sans nécessiter un matériel dédié. Le protocole IRT n'est pas supporté par CPUM.

**NOTE :** L'utilisation de la communication IRT nécessite d'équiper le réseau concerné de commutateurs réseau spécifiques pour le protocole IRT.

### 10.3.1 Traitement des erreurs

En plus du traitement des erreurs au niveau du serveur Modbus déjà décrit (voir la section 8.4.5 Traitement des erreurs et la section 9.3 Traitement des erreurs), le comportement des cartes CPUM avec logiciel version 801 (ou ultérieure) a été amélioré pour permettre de documenter les erreurs de communication additionnelles.

#### 10.3.1.1 Erreurs de communication internes

Étant donné que la couche PROFINET utilise le serveur Modbus existant pour accéder aux cartes du rack VM600 (voir la Figure 10-1), les erreurs de communication interne enregistrées seront comme décrit à la section 8.4.5 Traitement des erreurs et la section 9.3 Traitement des erreurs.

#### 10.3.1.2 Erreurs de communication externes

Les types de traitement des erreurs implémentés dépendront de l'automate PROFINET et du logiciel (DCS) utilisé.

**NOTE :** Les tentatives de communication (d'accès) à une carte non branchée donneront lieu à une erreur de communication retournée par DCS.

Par exemple SIMATIC STEP 7 utilise l'icône de données indisponibles lorsque qu'il n'est pas en mesure d'obtenir des données à afficher valides d'un objet (comme un VM600). L'utilisation de cette icône est représentée à titre d'exemple dans la colonne « Status value » de la Figure 10-2. (À titre de comparaison, la Figure 10-17 représente la même fenêtre, mais cette fois lorsque les données sont disponibles.)

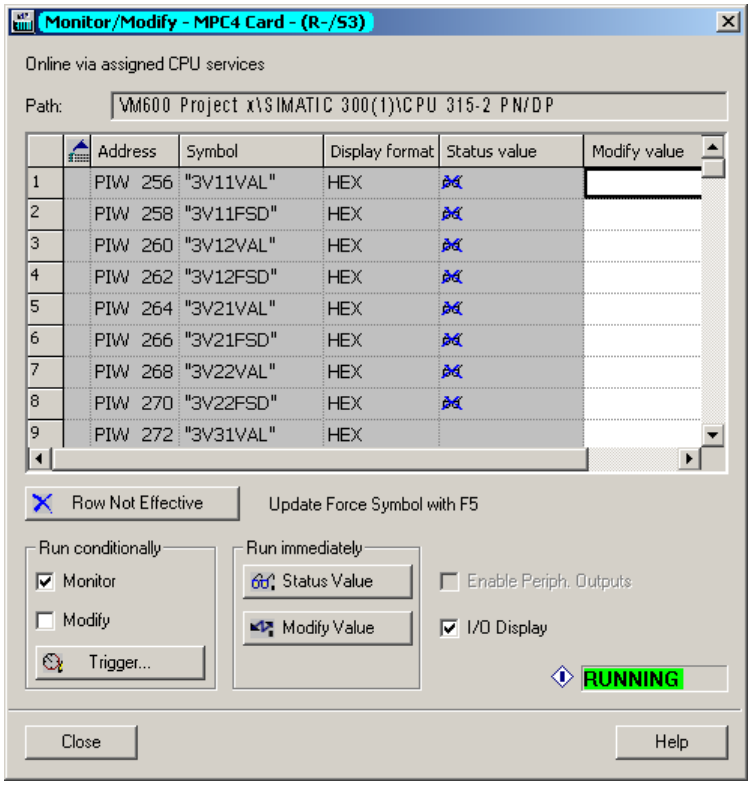



**Figure 10-2:** SIMATIC Manager – fenêtre « Monitor/Modify » avec données indisponibles

#### 10.3.1.3 Erreurs système

Voir 5.4.5.2 - Erreurs système.

## 10.4 PROFINET IO

PROFINET IO utilise un transfert cyclique des données pour l'échange de données avec des automates programmables via Ethernet. En fait, on peut le considérer comme un PROFIBUS sur Ethernet.

Les données échangées sont organisées par emplacements (slots) contenant des modules (avec des points I/O). L'automate et l'équipement avec lequel il communique doivent comprendre la structure et la signification des données échangées. Ceci s'effectue en utilisant des fichiers GSDML (voir 10.5 Fichiers GSDML).

PROFINET IO classe les équipements en fonction de leur rôle dans un système. Il existe trois possibilités :

- Automate IO
  Équipements dotés d'un programme d'automatisation pour contrôler un système ou un sous-système, par exemple un automate universel Siemens SIMATIC S7-300.
- Équipement IO
  Équipements distribués dans un système ou un sous-système et destinés à exécuter des fonctions de capteur ou d'actionneur, par exemple un rack avec des cartes VM600.
- Superviseur IO
  Un ordinateur ou une interface homme-machine utilisé typiquement pour faire exécuter ou superviser des opérations.

Un système PROFINET IO nécessite au moins un automate IO et un équipement IO. Les configurations comprenant un automate IO et plusieurs équipements IO sont les plus fréquentes (voir 10.7 Configuration d'un système VM600 pour fonctionner en tant qu'équipement E/S PROFINET IO).

## 10.5 Fichiers GSDML

L'intégration d'équipements extérieurs dans un système PROFINET nécessite de recourir à des fichiers GSDML. Ce concept est identique à celui utilisé pour PROFIBUS (fichiers GSD), à l'exception du fait que PROFINET utilise des fichiers système appelés fichiers GSDML, car sur base XML.

Le fichier GSDML est utilisé pour identifier les caractéristiques opérationnelles de base d'un équipement PROFINET et permettre l'interopérabilité et l'interchangeabilité. Ceci autorise l'utilisation d'outils de configuration indépendants de la marque des matériels. Typiquement, un fichier GSDML contient des informations sur le fournisseur, l'information temporelle, les options ou fonctionnalités implémentées ainsi qu'une liste des signaux E/S disponibles. Un fichier GSDML est nécessaire pour chaque équipement IO PROFIBUS.

Meggitt fournit des fichiers GSDML pour tous les équipements pouvant fonctionner sous PROFINET et destinés à décrire l'implémentation de l'équipement IO. Par exemple :

- GSDML-V2.1-VM600-CPUM-20100624.xml.

**NOTE :** Contacter Meggitt pour obtenir les derniers fichiers GSDML disponibles pour nos équipements E/S.

Voir 10.7.3.4 Ajouter un équipement IO vibro-meter sur SIMATIC STEP 7 pour la procédure d'installation (importation) d'un fichier GSDML au niveau du logiciel Siemens.

Dans le cas des données I/O, le fichier GSDML décrit la structure des données d'entrée/sortie cycliques transmises entre l'automate programmable et l'équipement IO.
Toute incohérence entre la taille ou la structure des données d'entrée/sortie et la structure interne actuelle de l'équipement donnera lieu à une alarme au niveau du contrôleur/automate.

## 10.6 Fichiers de configuration PROFINET

Dans la mesure où l'interface logicielle PROFINET s'appuie sur l'interface Modbus existante, le fichier de configuration Modbus existant (`modbusDefault.cfg`) est toujours utilisé, comme indiqué sur la Figure 10-3.

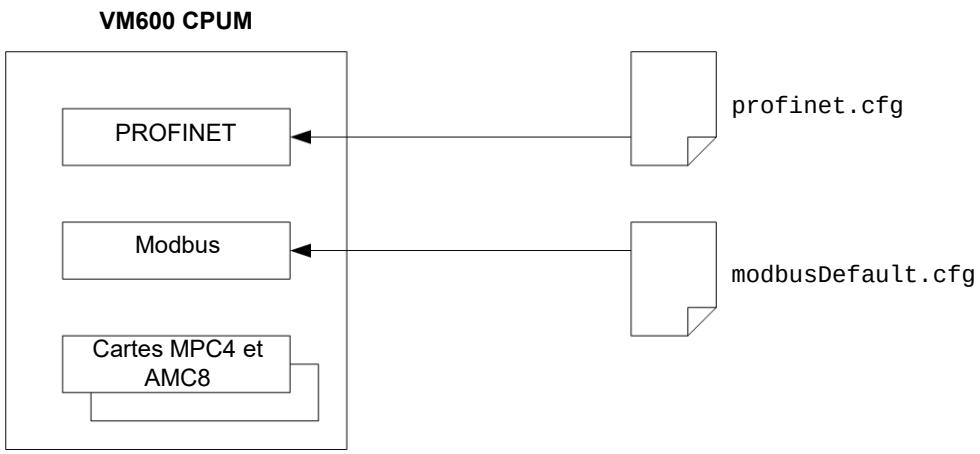



**Figure 10-3:** Configuration PROFINET de la carte VM600 CPUM

Cependant, pour prendre en charge la couche PROFINET supplémentaire, un fichier de configuration PROFINET supplémentaire (`profinet.cfg`) est également requis, comme indiqué sur la Figure 10-3.

Pour plus d'informations sur le fichier de configuration Modbus, tel qu'utilisé pour PROFINET, voir la section 10.6.1.1 Fichier modbusDefault.cfg. Pour plus d'informations sur le fichier de configuration PROFINET, voir la section 10.6.1.2 Fichier profinet.cfg.

Pour des informations générales sur l'utilisation des fichiers de configuration Modbus et PROFINET, voir les sections 2.5 Utilisation des fichiers de configuration et 2.6 Configuration des cartes CPUM.

### 10.6.1 Cartes CPUM exécutant la version 081 du microprogramme

#### 10.6.1.1 Fichier modbusDefault.cfg

La Figure 10-4 est un extrait d'un fichier de configuration `modbusDefault.cfg` type. Lorsqu'il est utilisé pour les communications PROFINET, ce fichier de configuration est essentiellement le même que pour Modbus, mais il est étendu avec des balises PROFINET supplémentaires. Les paramètres de communication sont définis dans les sections [RTUx], [TCP] et [GLOBAL] de ce fichier de configuration.

```
[GLOBAL]
DEFAULT_LONG_ORDER    = LL
DEFAULT_FLOAT_ORDER   = FL
IS_FULLY_COMPATIBLE   = NO  // YES/NO (default N)
IS_PROFINET_ACTIVATED = YES
```

**Figure 10-4:** Extrait d'un fichier de configuration `modbusDefault.cfg` type utilisé pour les communications PROFINET et incluant les paramètres de communication

Comme indiqué sur la Figure 10-4, pour les communications PROFINET, PROFINET est activé dans la section [GLOBAL] à l'aide de la balise IS_PROFINET_ACTIVATED :

- Si cette balise prend la valeur YES (condition vraie), la prise en charge de PROFINET est activée
- Si cette balise prend la valeur NO (condition fausse), la prise en charge de PROFINET n'est pas activée
- Si cette balise est manquante (non définie), la prise en charge de PROFINET n'est pas activée (c'est-à-dire que la valeur par défaut est NO).

Comme indiqué sur la Figure 10-5, pour les communications PROFINET, les emplacements PROFINET sont définis à la fin de la section [MAPPING] à l'aide de la balise PROFINET_CREATE. Un registre au moins doit être défini par cette balise. Cela signifie que la taille minimale d'un emplacement PROFINET est de 2 octets, et sa taille maximale de 128 octets.

En général, la taille de l'emplacement doit être une puissance de deux. Il existe cependant deux exceptions à cette règle :

- La carte MPC4 possède une taille d'emplacement de 74 octets
- La carte AMC8 possède une taille d'emplacement de 102 octets.

```
[MAPPING]

 ...

PROFINET_CREATE Slot3 0 36
PROFINET_CREATE Slot4 64 100
PROFINET_CREATE Slot5 128 164
PROFINET_CREATE Slot6 192 228
PROFINET_CREATE Slot7 256 292
PROFINET_CREATE Slot8 320 356
PROFINET_CREATE Slot9 384 420
PROFINET_CREATE Slot10 448 484
PROFINET_CREATE Slot11 512 548
PROFINET_CREATE Slot12 576 612
PROFINET_CREATE Slot13 640 676
PROFINET_CREATE Slot14 704 740
```

**Figure 10-5:** Extrait d'un fichier de configuration `modbusDefault.cfg` type utilisé pour les communications PROFINET et incluant les emplacements PROFINET

Comme indiqué sur la Figure 10-5, la balise PROFINET_CREATE utilise le numéro d'emplacement de rack VM600 (3 à 14), suivi des adresses de début et de fin des registres Modbus où les données de la carte sont mappées, en fonction de la configuration Modbus.

Pour les communications PROFINET, les registres MPC4 doivent être définis comme indiqué dans le Tableau 10-1 et les registres AMC8 doivent être définis comme indiqué dans le Tableau 10-2, car ces tableaux (définitions d'emplacements PROFINET) correspondent aux structures de données des emplacements PROFINET sous-jacentes définies dans un fichier GSDML (voir la section 10.5 Fichiers GSDML).

**Tableau 10-1** : Format de définition du registre MPC4

| Type de canal | Format de registre requis |
|---|---|
| Canal | Valeur sortie1, FSD, Statut (Invalid, A +, A–, D +, D–, SOK)<br>Valeur sortie2, FSD, Statut (Invalid, A +, A–, D +, D–, SOK) |
| Double canal | Valeur, FSD, Statut (Invalid, A +, A–, D +, D–, SOK) |
| Vitesse | Valeur, Statut (Invalid, A+, A−, SOK) |
| Fonctions de base et avancées | BF1 à BF8, AF1 à AF4 |
| Statut commun | FMC, FCR, CPR, CIE, CA, CD, CSOL, CTL, CTOR, CDSE, CISE, CCME, SL, TM, DBP, AR |
| Statut de la carte | MCR (configuration MPC en cours d'exécution) |

**Tableau 10-2** : Format de définition du registre AMC8

| Type de canal | Format de registre requis |
|---|---|
| Canal | Valeur sortie1, FSD, Statut (A–, A +, D–, D +, GOKF, ADERR, ADSTBY, ADLOCK, ADTXERR, ADCFGERR, BITF, NOSPL, OKF, LINERR, JERR) |
| Canaux multiples | Valeur, FSD, Statut (A–, A +, D–, D +, GOKF, ADERR, ADSTBY, ADLOCK, ADTXERR, ADCFGERR, BITF, NOSPL, OKF, LINERR, JERR) |
| Fonction de base | BF1 à BF16 |
| Fonction avancée | AF1 à AF8 |
| Alarme canal et canaux multiples utilisée | A− utilisée, A+ utilisée, D− utilisée, D+ utilisée |
| Statut de la carte | CSANR, CSSL, CSDB, CSAR |

#### 10.6.1.2 Fichier profinet.cfg

La Figure 10-6 représente un fichier de configuration `profinet.cfg` type. Ce fichier de configuration définit le nom du périphérique et la passerelle utilisés pour les communications PROFINET.

**Figure 10-6:** Fichier `profinet.cfg` type

## 10.7 Configuration d'un système VM600 pour fonctionner en tant qu'équipement E/S PROFINET IO

Cette section donne un exemple de communications PROFINET avec un rack VM600 utilisant le logiciel Siemens SIMATIC STEP 7 comme exemple de DCS PROFINET (voir la Figure 10-7). Cet exemple illustre les concepts requis pour l'interfaçage d'un rack VM600 avec tout DCS prenant en charge PROFINET.

**NOTE :** Pour que les communications PROFINET fonctionnent, les informations définies dans les fichiers de configuration PROFINET (`modbusDefault.cfg` et `profinet.cfg`) sur la carte CPUM doivent correspondre aux informations définies dans le DCS (tel que Siemens SIMATIC STEP 7).

### 10.7.1 Équipement utilisé

#### 10.7.1.1 Matériel

Un rack VM600 contenant les cartes suivantes :

- Carte CPUM – avec logiciel version 801 ou ultérieure
- Carte IOCN
- Carte ou cartes MPC4
- Carte ou cartes AMC8.

Automate universel Siemens SIMATIC S7-300 (appelé également CPU ou API/PLC) :

- CPU315-2 PN/DP (315-2EH13-0AB0) avec PSU externe.

Ordinateur sous Microsoft® Windows XP.

Commutateur Netgear® 10/100 Mbps Fast Ethernet :

- FS105.

#### 10.7.1.2 Logiciel

La carte CPUM doit fonctionner avec le logiciel version 801 ou ultérieure.

Siemens SIMATIC STEP 7 (la version 5.4 service pack 5 a été utilisée).

Nous vous conseillons de consulter le site Internet Siemens pour les dernières informations mises à jour sur le logiciel SIMATIC STEP 7 et les automates universels :

http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/Pages/Default.aspx

http://www.automation.siemens.com/mcms/programmable-logic-controller/en/simatic-s7-controller/Pages/Default.aspx

### 10.7.2 Conditions préalables

S'assurer que tout l'équipement (partie matérielle) est connecté comme décrit dans le réseau local de la Figure 10-7.

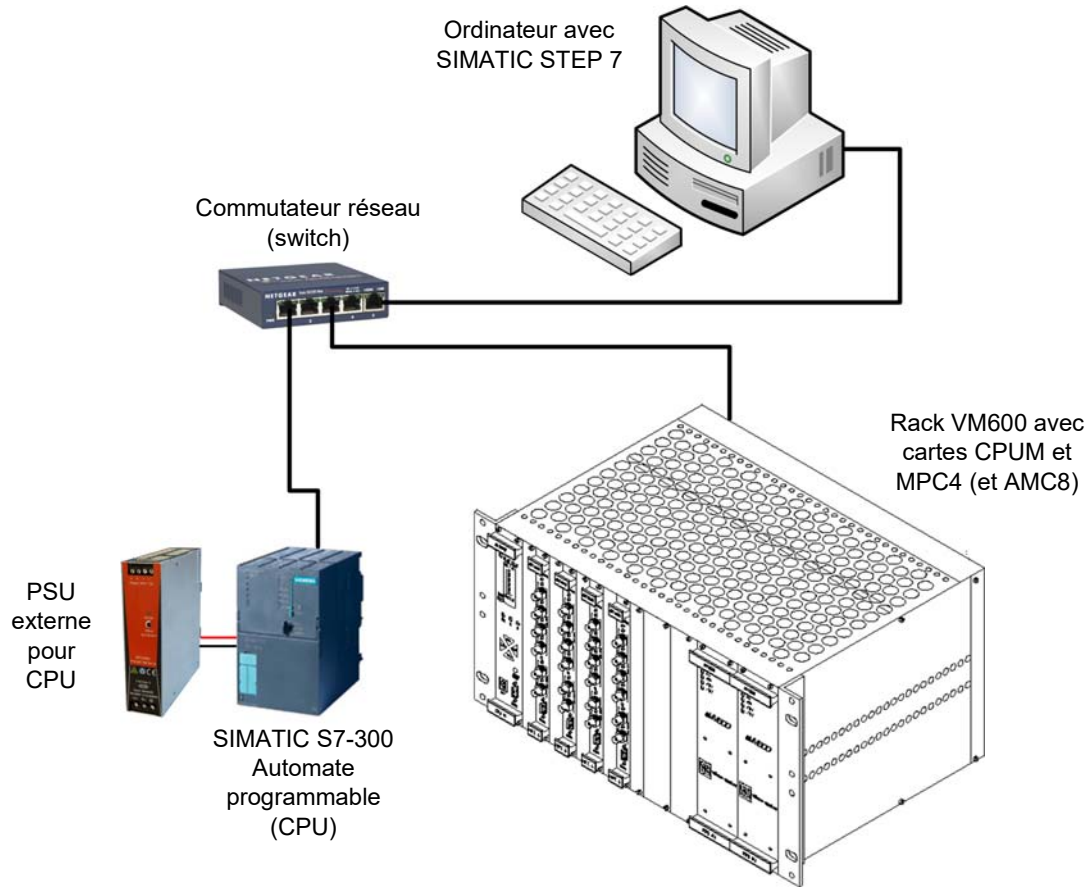



**Figure 10-7:** Exemple réseau PROFINET

### 10.7.3 Procédure de configuration d'un réseau PROFINET avec SIMATIC STEP 7

Un certain nombre d'étapes de configuration et d'activation du logiciel SIMATIC STEP 7 sont nécessaires avant de pouvoir parvenir à une communication avec un VM600 via PROFINET. Le détail des étapes nécessaires peut différer selon le fournisseur et la version du logiciel DCS utilisé.

#### 10.7.3.1 Démarrer le logiciel SIMATIC et créer un nouveau projet

**1-** Cliquer sur **Start > SIMATIC > SIMATIC Manager** ou utiliser un raccourci du bureau si disponible.

**NOTE :** Si l'assistant STEP 7 « new project » apparaît, cliquer sur Annuler pour pouvoir configurer le projet manuellement.

La fenêtre SIMATIC Manager apparaît.

**2-** Cliquer sur **File > New**.

La boîte de dialogue « New Project » apparaît.

**3-** Saisir un nom approprié pour le projet (par exemple *VM600_PROFINET*) et s'assurer que le type est bien un projet. Cliquez sur **OK**.
SIMATIC Manager créé le nouveau projet

### 10.7.3.2 Ajouter des objets à un projet

Le logiciel SIMATIC fait appel au concept d'objets – chaque entité ajoutée à un projet SIMATIC est considérée comme un objet, y compris l'automate Siemens et le matériel VM600.

**1-** Cliquer avec le bouton droit de la souris sur le projet (par exemple *VM600_PROFINET)* et sélectionner **Insert New Object > SIMATIC 300 Station** pour ajouter un automate CPU au projet. (Il est possible d'utiliser également le menu Insert pour ajouter des objets à un projet.)
Un objet SIMATIC 300 CPU est ajouté au projet avec le nom SIMATIC 300(1) par défaut.

**2-** Cliquer avec le bouton droit de la souris sur le projet et sélectionner **Insert New Object > Industrial Ethernet** pour ajouter Ethernet industriel/PROFINET au projet.
Un objet Ethernet est ajouté au projet avec le nom Ethernet(1) par défaut.

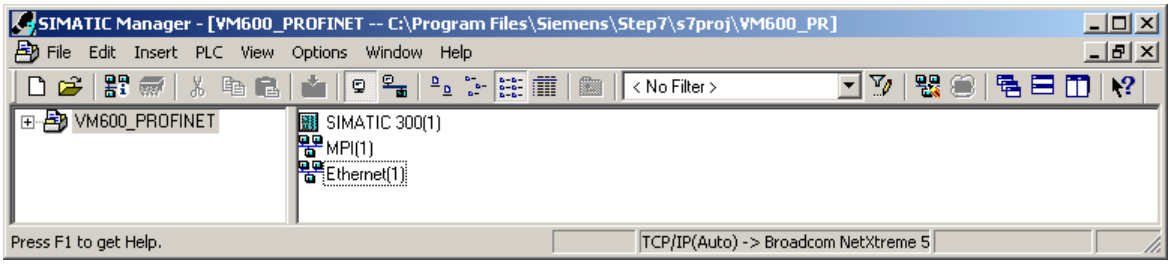



**Figure 10-8:** SIMATIC Manager – ajouter des objets à un projet

**3-** Cliquer avec le bouton droit de la souris sur l'objet MPI(1) dans la fenêtre principale et sélectionner **Delete** pour supprimer le réseau MPI. (Celui-ci a été ajouté par défaut lors de la création du projet, mais il n'est pas nécessaire.)

Le projet contient maintenant deux des trois objets requis par ce système PROFINET, c'est-à-dire une « station PROFINET » (l'automate CPU) et un réseau Ethernet industriel/PROFINET. Le troisième composant, en l'occurrence le matériel VM600, sera ajouté ultérieurement après avoir enregistré les fichiers GSML fournis par Meggitt dans le logiciel SIMATIC (voir 10.7.3.4 Ajouter un équipement IO vibro-meter sur SIMATIC STEP 7).

### 10.7.3.3 Configurer l'automate SIMATIC CPU

Les objets ajoutés au projet SIMATIC, en l'occurrence l'automate universel Siemens SIMATIC S7-300 (CPU), doit maintenant être configuré.

**1-** Cliquer avec la touche droite de la souris sur l'objet SIMATIC 300(1) puis cliquer sur **Open Object**.

La fenêtre de configuration du matériel « HW Config » apparaît et affiche la configuration actuelle de la station PROFINET SIMATIC 300.

La fenêtre « HW Config » est subdivisée en trois fenêtres : une fenêtre pour la station (en haut à gauche), une fenêtre pour le catalogue Hardware (à droite) et une fenêtre en ligne (en bas à gauche). Noter que la fenêtre pour la station est vide, ce qui signifie que le matériel n'est pas configuré.

**NOTE :** Pour afficher la fenêtre "HW Config" à n'importe quel moment, cliquer avec la touche de droite de la souris sur l'objet SIMATIC 300(1) puis cliquer sur **Open Object**.

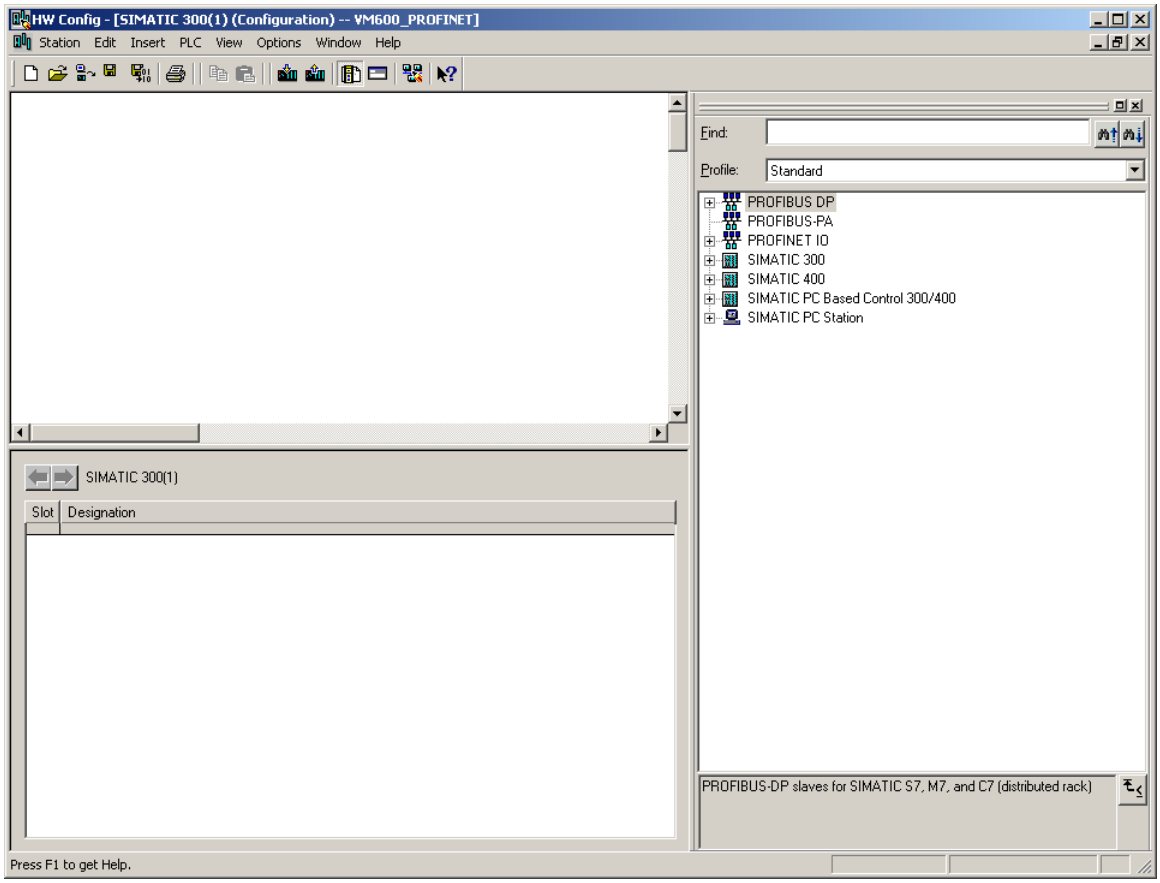



**Figure 10-9:** HW Config – avec objets non configurés (la fenêtre de la station est vide)

**2-** Dans la fenêtre de la station, faire un clic droit avec la souris et cliquer sur **Insert Object**. Puis cliquer sur **SIMATIC 300**, **RACK-300** et **Rail** dans les boîtes de dialogue qui apparaissent.
Un objet « Rail » sera ajouté au projet (dans la fenêtre de la station) avec le nom (0)UR par défaut.

**3-** Dans la fenêtre du catalogue Hardware, taper la version de votre CPU (automate) dans la boîte « Find », par exemple 315-2EH13-0AB0, puis appuyer sur la touche ENTRÉE.

Le logiciel SIMATIC va rechercher puis lister toutes les versions du microprogramme (firmware) pour votre CPU.

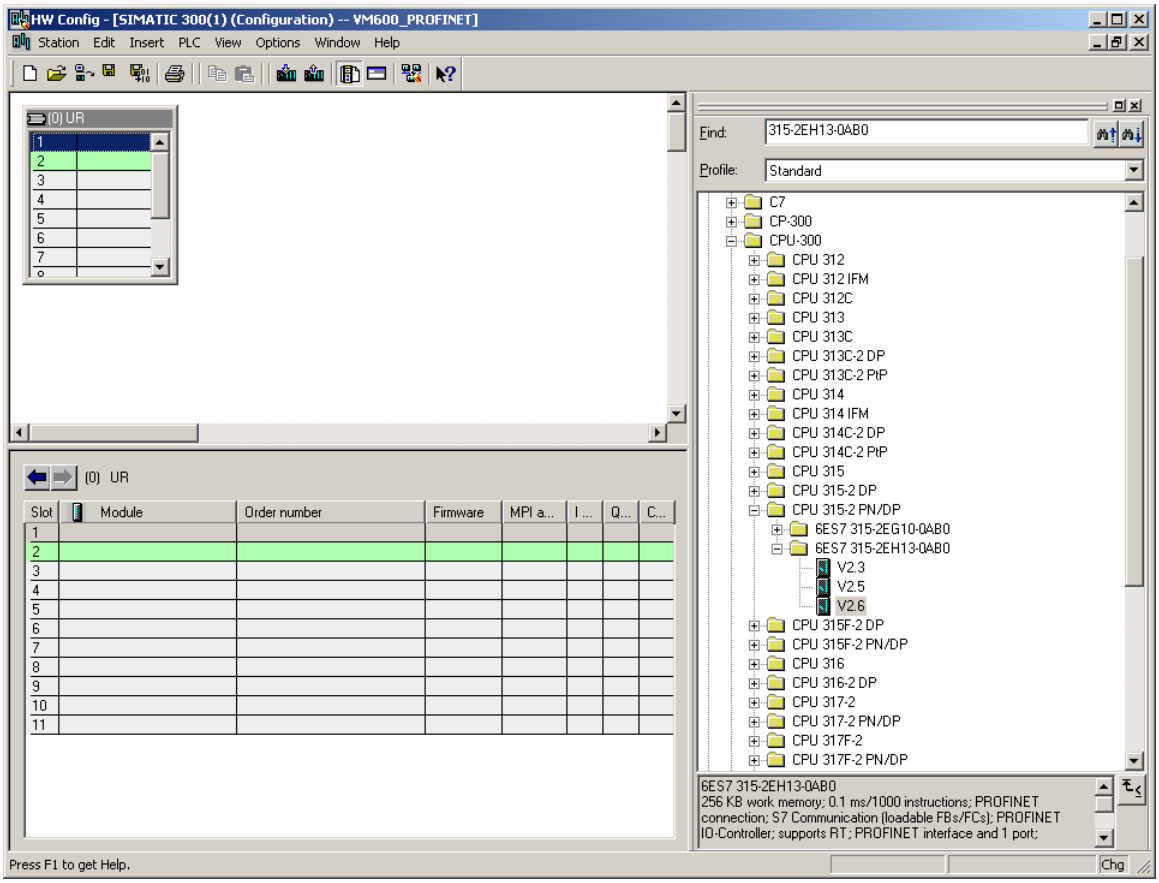



**Figure 10-10:** HW Config – la fenêtre catalogue Hardware avec les CPU firmware disponibles

**4-** Pour associer la version appropriée du microprogramme CPU à votre projet, utiliser l'une des solutions ci-dessous :
Double-clic sur la version appropriée du microgramme
Ou déplacer la version appropriée du microgramme sur le second slot du rail.

**5-** Les propriétés – La boîte de dialogue Ethernet interface PN-IO apparaît.
Entrer l'information appropriée du réseau Ethernet utilisé. À titre d'exemple, pour le réseau local privé défini à la Figure 10-7, l'information appropriée peut être :
10.10.52.62 pour l'adresse IP
255.255.255.0 pour le masque du sous-réseau
Et « Do not use router » était sélectionné.
Puis sélectionner Ethernet(1) dans la zone sous-réseau et cliquer sur **OK**.

Un système PROFINET IO est ajouté à l'objet Rail (dans la fenêtre de la station).

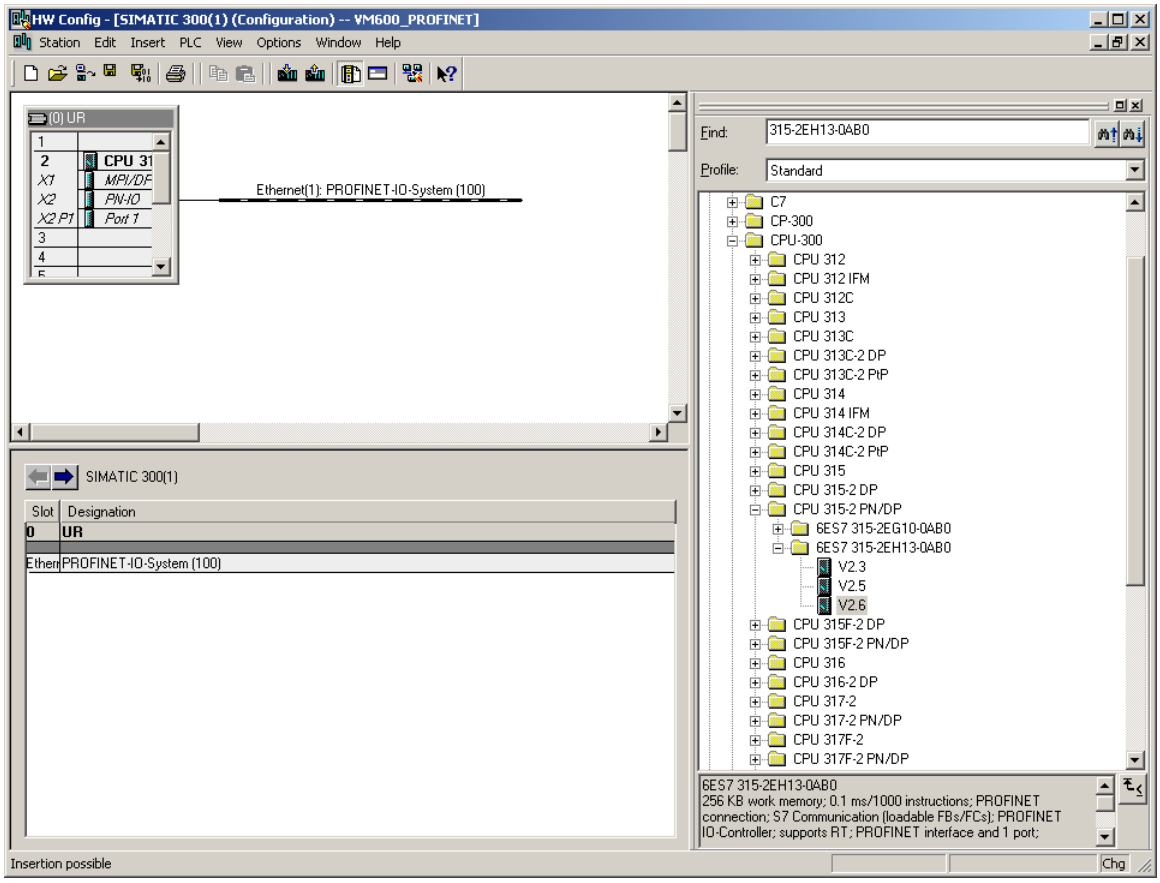



**Figure 10-11:** HW Config – objets configurés dans la fenêtre Station

### 10.7.3.4 Ajouter un équipement IO vibro-meter sur SIMATIC STEP 7

Avant d'ajouter un projet vibro-meter® au projet SIMATIC STEP 7, il est nécessaire au préalable d'ajouter l'équipement E/S (« IO-Device ») en tant qu'objet au catalogue Hardware. Ceci se fait par l'intermédiaire des fichiers GSDML fournis par Meggitt (voir 10.5 Fichiers GSDML).

**1-** Cliquer sur **Options > Install GSD File**.

Dans la fenêtre d'installation des fichiers GSD qui apparaît, vérifier que l'option sélectionnée pour « Install GSD Files » est bien « from the directory ». Cliquer sur le bouton « Browse » et chercher le dossier où sont stockés les fichiers GSDML. Cliquez sur **OK**.

La boîte de dialogue affiche alors une liste de tous les fichiers GSD disponibles dans le dossier.

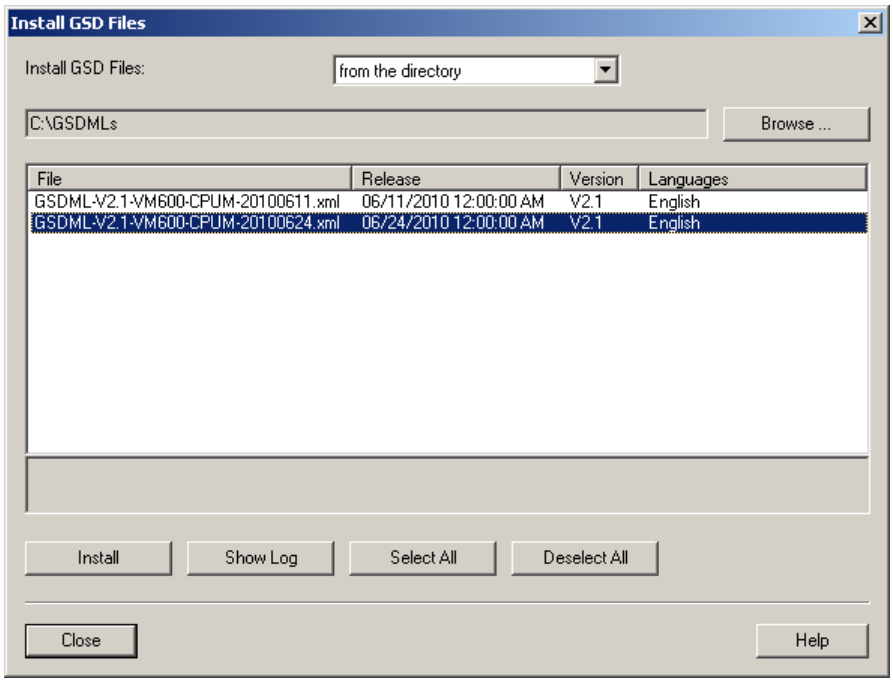



**Figure 10-12:** HW Config – installation des fichiers GSDML

**2-** Sélectionner le fichier GSD approprié dans la liste et cliquer sur **Install**. Puis cliquer sur **Yes** pour confirmer l'installation du GSD.

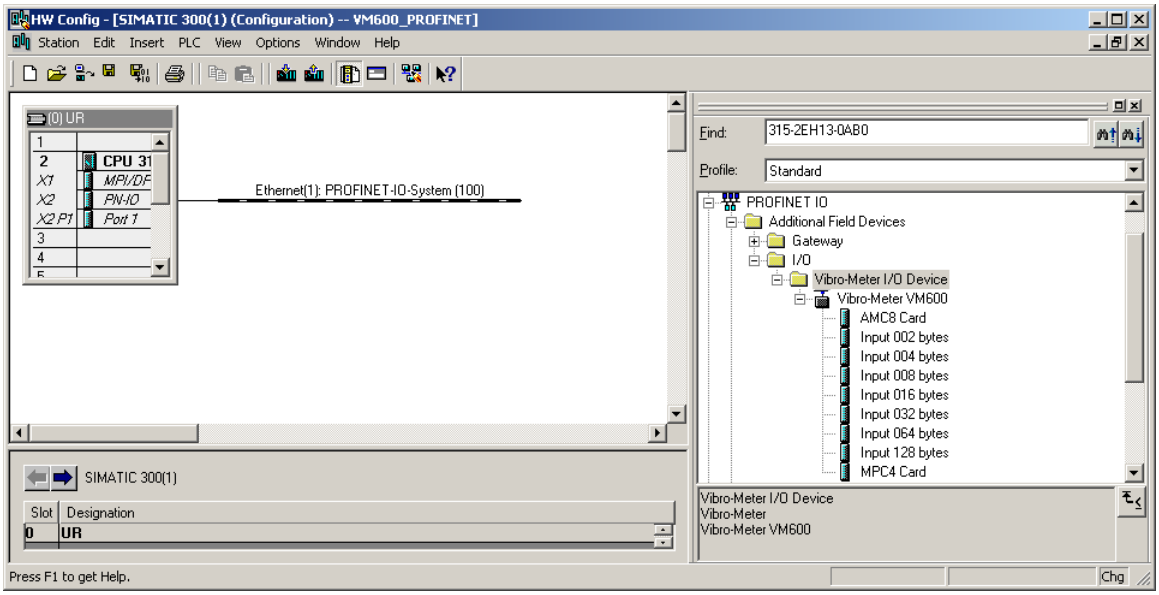



**Figure 10-13:** HW Config – avec rack et cartes VM600 disponibles

L'équipement vibro-meter® est ajouté au logiciel SIMATIC STEP 7 et peut être trouvé dans le catalogue Hardware sous : PROFINET IO – Additional Field Devices – I/O – Vibro-Meter I/O Device – Vibro-Meter VM600.

Le fait d'installer un fichier GSDML (.xml) ajoute une copie du fichier GSDML au référentiel GSD et aux fichiers GSDML que le logiciel Siemens SIMATIC tient à jour sur

l'ordinateur. L'emplacement de ce dossier va dépendre de l'endroit où vous avez installé le logiciel SIMATIC.
Par exemple : `C:\Program Files\Siemens\Step7\S7DATA\GSD`.

**NOTE :** Les fichiers GSDML ne peuvent pas être ajoutés depuis ce dossier. Il s'agit d'un référentiel des versions, tenu à jour par le logiciel SIMATIC.

### 10.7.3.5 Ajouter un rack VM600 au projet

Comme le rack Vibro-Meter, de Meggitt, existe maintenant dans le catalogue Hardware (voir 10.7.3.4 Ajouter un équipement IO vibro-meter sur SIMATIC STEP 7), il peut être ajouté au réseau PROFINET (qui a été configuré à l'étape 10.7.3.2 Ajouter des objets à un projet).

**1-** Cliquer sur l'icône de rack Vibro-Meter VM600 dans le catalogue Hardware et le déplacer vers l'objet Ethernet(1) pour ajouter un rack au réseau PROFINET.

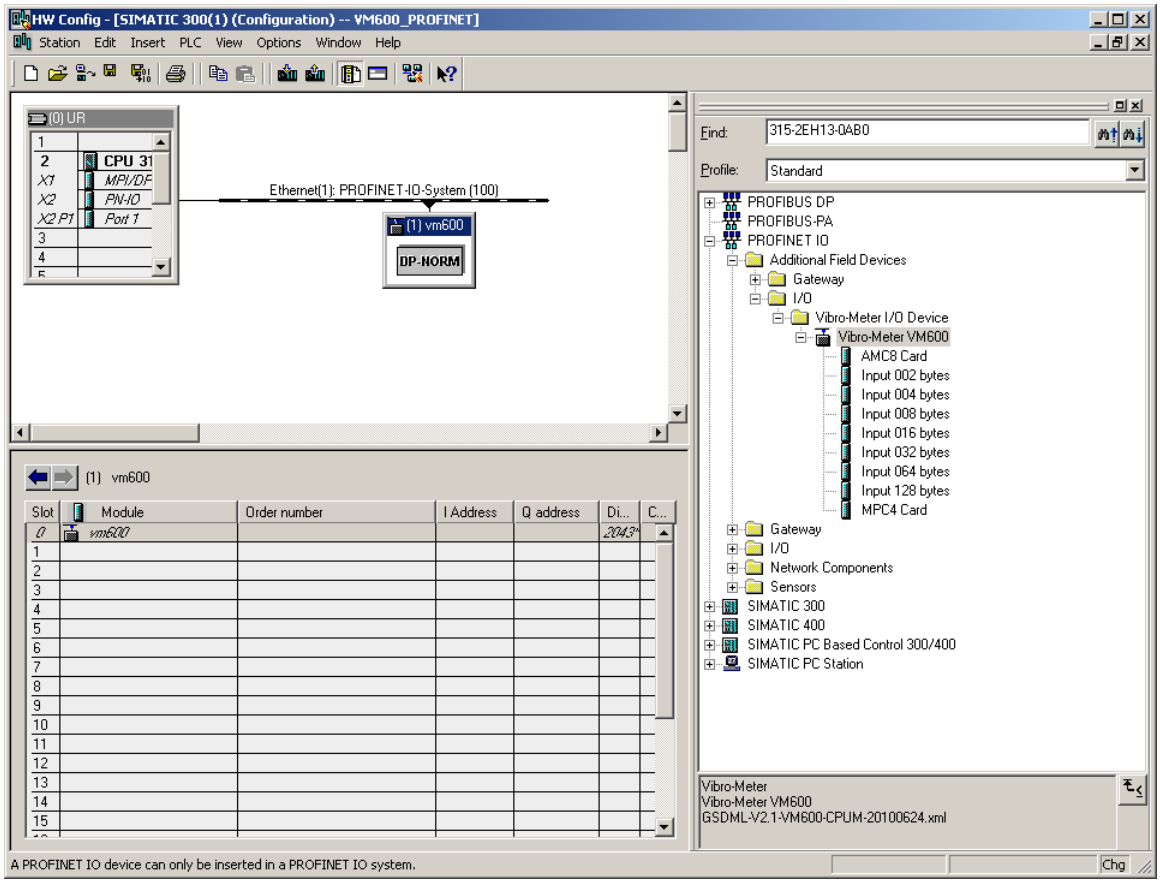



**Figure 10-14:** HW Config – avec un rack VM600 ajouté au projet

La fenêtre en ligne (en bas à gauche) se modifie et fait apparaître les slots disponibles à l'intérieur du rack VM600. Pour un nouveau rack, les slots 3 à 14 sont disponibles pour accepter des cartes (mais tous les slots 0 à 24 s'affichent).

### 10.7.3.6 Ajouter une carte MPC4 ou AMC8 au projet

Comme les cartes VM600 Vibro-Meter, de Meggitt, existent maintenant dans le catalogue Hardware, elles peuvent être ajoutées au réseau PROFINET (qui a été configuré à l'étape 10.7.3.2 Ajouter des objets à un projet).

**NOTE :** Par défaut, l'équipement Vibro-Meter VM600 n'a aucune carte installée, car il est impossible de gérer des fichiers GSD pour toutes les configurations possible de racks et de cartes VM600. Il est par conséquent nécessaire d'ajouter les cartes MPC4 individuelles et les objets cartes AMC8 au rack VM600 pour pouvoir refléter votre matériel actuel.

**1-** Cliquer sur l'icône de carte Vibro-Meter VM600 – soit MPC4 soit AMC8 – dans le catalogue Hardware et la déplacer vers le slot de la fenêtre en ligne qui correspond à votre logiciel rack VM600. (Les slots pouvant accepter une carte deviennent verts quant une carte est sélectionnée dans la catalogue Hardware.)

Ceci ajoutera un objet carte à l'objet rack VM600 dans le projet SIMATIC.

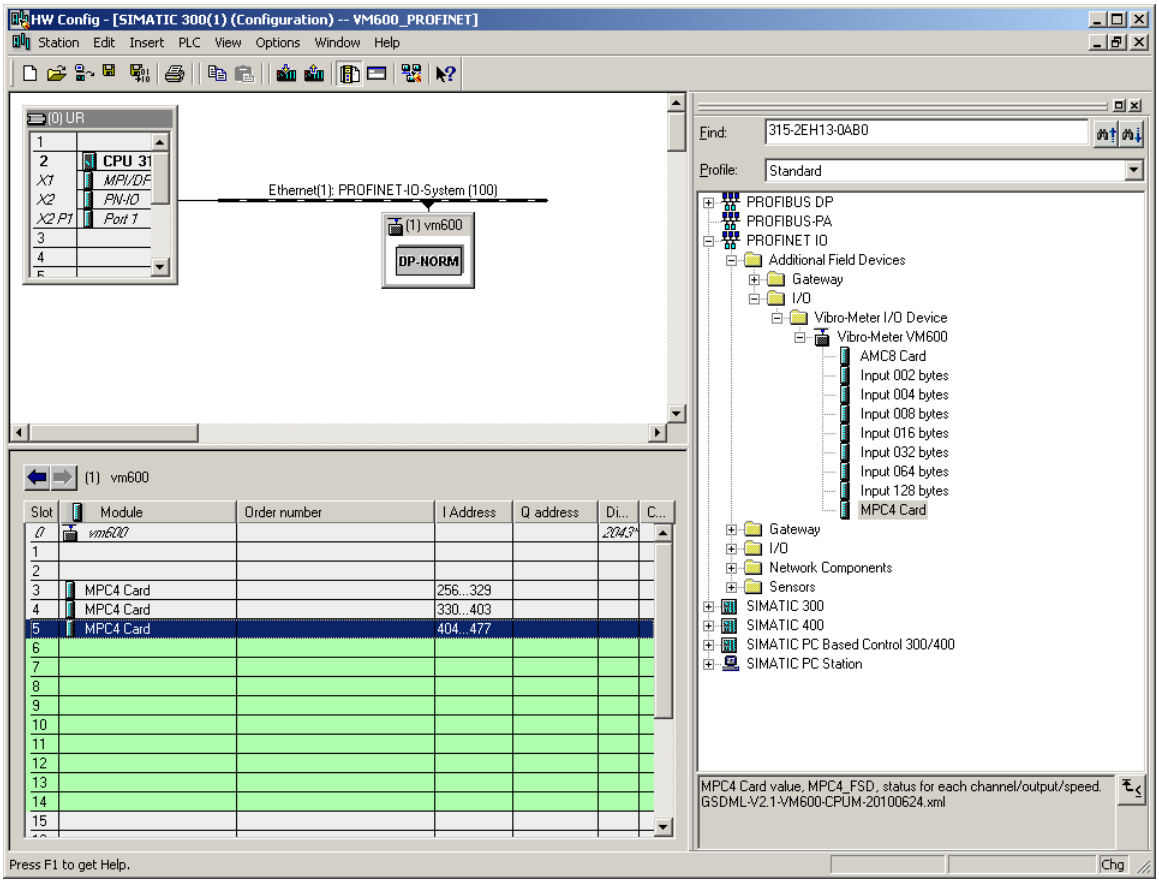



**Figure 10-15:** HW Config – avec cartes VM600 ajoutées au projet

Répéter cette étape jusqu'à ce que le modèle du matériel VM600 (rack et cartes) créé dans le projet SIMATIC corresponde au matériel VM600 utilisé actuellement.

**NOTE :** Il est important que le nombre de slots utilisés dans le projet SIMATIC soit identique au nombre utilisé réellement au niveau matériel, car ce nombre sert à calculer les adresses des variables.

Vous pouvez cliquer avec la touche de droite de la souris sur n'importe quelle rangée de cartes VM600 (slot) contenant une carte puis cliquer sur **Delete** pour supprimer une carte non souhaitée.

### 10.7.3.7 Contrôler les propriétés d'un objet SIMATIC

La fenêtre des propriétés des objets peut être utilisée à tout moment pour visualiser (ou modifier) les paramètres des objets ajoutés à un projet.

**1-** Pour accéder à la fenêtre des propriétés d'un objet, soit :
Double clic sur l'objet
Soit clic droit sur l'objet et clic sur **Object Properties**.

Par exemple, double-cliquer sur l'objet rack VM600 dans la fenêtre de la station (en haut à gauche) pour ouvrir la fenêtre Properties – vm600 et contrôler la configuration du rack VM600.

Par exemple, cliquer sur la rangée de cartes VM600 dans la fenêtre en ligne (en bas à gauche) et cliquer sur **Object Properties** pour ouvrir la fenêtre Properties – MPC4 Card (ou Properties – AMC8 Card) et contrôler la configuration de la carte VM600.

Bien que le nom d'équipement donné dans l'objet rack VM600 puisse être modifié en utilisant le logiciel SIMATIC, il doit être le même que le nom défini dans le fichier de configuration profinet.cfg (voir 10.6 Fichiers de configuration PROFINET). C'est aussi le nom d'équipement du rack VM600 enregistré sur la carte CPUM.

**NOTE :** Chaque équipement utilisé dans le réseau doit avoir un nom unique. Le nom d'équipement est restreint aux caractères minuscules uniquement.

Le projet défini dans SIMATIC STEP 7 HW Config peut être enregistré à tout moment par l'une des méthodes suivantes :

Clic sur **Station > Save**.

Ou clic sur l'icône de la barre d'outils.

### 10.7.3.8 Configurer l'interface PG/PC

L'interface PG/PC est utilisée par le logiciel SIMATIC STEP 7 pour programmer des équipements raccordés à l'ordinateur, par exemple SIMATIC S7-300 CPU (automate). Cette interface doit être configurée correctement avant de télécharger le projet vers le CPU.

L'interface PG/PC fait partie du SIMATIC Manager et non du SIMATIC HW Config.

**1-** Dans SIMATIC Manager, cliquer sur **Options > Set PG/PC Interface**.

La fenêtre « PG/PC Interface » apparaît.

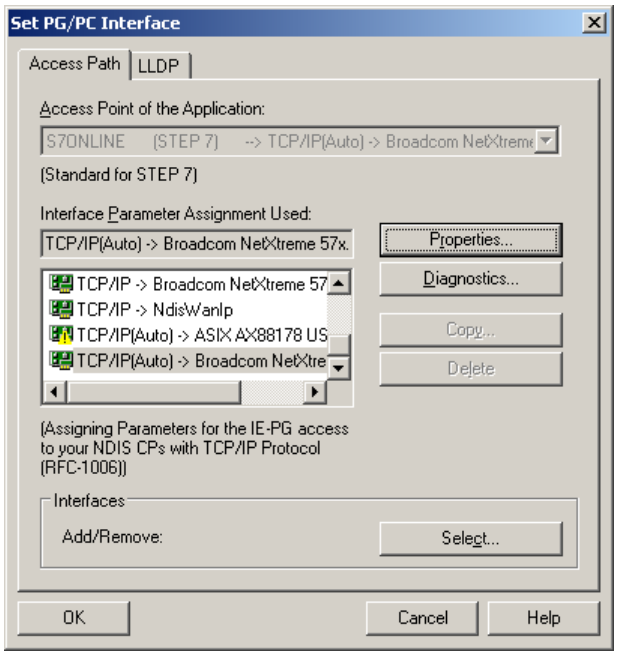



**Figure 10-16:** SIMATIC Manager – la fenêtre « Set PG/PC Interface »

**2-** Sélectionner la ligne TCP/IP(Auto) dans la liste « Interface Parameter Assignment Used » correspondant à la carte d'interface réseau de l'ordinateur que vous utilisez pour programmer SIMATIC CPU.

Choisir et implémenter le chemin d'accès qui correspond le mieux à votre environnement. Dans cet exemple, on utilise le chemin d'accès TCP/IP (Auto).

**3-** Dans la fenêtre « Properties – TCP/IP(Auto) » qui apparaît, vérifier que l'option « Do not assign IP addresses automatically » est sélectionnée. (Cliquer sur OK puis encore une fois sur OK pour revenir.)

### 10.7.3.9 Télécharger la configuration vers SIMATIC CPU

La configuration du projet peut maintenant être envoyée sur le SIMATIC CPU.

Le téléchargement est lancé depuis SIMATIC HW Config, pas depuis SIMATIC Manager.

**1-** Dans SIMATIC HW Config, cliquer sur le bouton de téléchargement vers le module de la barre d'outils.

**2-** Dans la boîte de dialogue du projet « Message Number Assignment Selection » qui apparaît, sélectionner l'option « Assign CPU-oriented unique message numbers » puis cliquer sur **OK**.

**3-** Dans la boîte de dialogue « Select Target Module » qui apparaît, sélectionner le module cible qui correspond à votre matériel puis cliquer sur **OK**.
Par exemple CPU 315-2 PN/DP (voir 10.7.1 Équipement utilisé).

**4-** Dans la boîte de dialogue « Select Node Address » qui apparaît, s'assurer que l'adresse IP soit identique avec celle de la carte CPUM dans le rack VM600. Par exemple 10.10.52.20. (Cette adresse IP doit être la même que celle configurée au chapitre 10.7.3.3 Configurer l'automate SIMATIC CPU.)

---

**NOTE :** Pour envoyer la configuration à un autre nœud que le nœud par défaut, cliquer sur **View** pour rechercher les autres noeuds présents sur le réseau.

---

**5-** Le téléchargement démarre.

Le logiciel SIMATIC affiche une boîte de dialogue avant de stopper le module (automate CPU 315-2 PN/DP) au cours du processus de téléchargement. Cliquer sur OK pour continuer ou sur Annuler pour stopper la procédure.

Le logiciel SIMATIC affiche également une boîte de dialogue avant de redémarrer le module au cours du processus de téléchargement. Cliquer sur Yes pour continuer ou sur No pour stopper la procédure.

### 10.7.4 Procédures pour utiliser un réseau PROFINET avec SIMATIC STEP 7

Maintenant que le réseau PROFINET (Ethernet industriel) est configuré, le logiciel SIMATIC STEP 7 peut être utilisé pour communiquer avec le SIMATIC CPU (c'est-à-dire automate CPU 315-2 PN/DP) et effectuer des opérations sur le système, notamment :

- Gestion du mode CPU (comme démarrage et arrêt)
- Visualisation de l'information sur l'automate.

#### 10.7.4.1 Accès à l'automate

Ces opérations s'exécutent en utilisant SIMATIC Manager.

**1-** Cliquer sur **PLC > Diagnostic/Setting > Operating Mode** (ou CTRL+I) pour visualiser le mode de fonctionnement actuel du CPU.

La fenêtre « Operating Mode » qui apparaît peut être utilisée pour allumer et éteindre l'automate en utilisant les boutons STOP et « Warm Restart » (si supportés par votre automate).

**2-** Cliquer sur **PLC > Diagnostic/Setting > Module Information** (ou CTRL+D) pour visualiser l'information concernant le CPU.

La fenêtre "Module Information" qui apparaît fournit de nombreuses informations concernant l'automate. Par exemple, la liste d'événements dans le tableau « Diagnostic Buffer property » qui comptabilise les événements (avec l'heure d'occurrence) survenus sur ce module, comme l'arrêt ou le démarrage ou encore les erreurs module.

#### 10.7.4.2 Surveiller des équipements

Ces opérations s'exécutent en utilisant SIMATIC HW Config.

Le fait de sélectionner un objet dans la fenêtre de la station (en haut à gauche) met à jour l'information s'affichant dans la fenêtre en ligne (en bas à gauche).

**1-** Dans SIMATIC HW Config, cliquer sur l'objet (o)UR dans la fenêtre de la station. Le contenu de la fenêtre en ligne s'actualise pour fournir l'information relative à l'objet SIMATIC Rail.

**2-** Dans SIMATIC HW Config, cliquer sur l'objet (1)vm600 dans la fenêtre de la station. Le contenu de la fenêtre en ligne s'actualise pour fournir l'information relative à l'objet rack VM600. Les cartes ajoutées lors de la configuration apparaissent également.

**3-** Faire un clic droit sur une carte MPC4 (ou AMC8) dans un slot sur la fenêtre en ligne puis cliquer sur **Monitor/Modify**.

La fenêtre « Monitor/Modify » pour la carte apparaît.

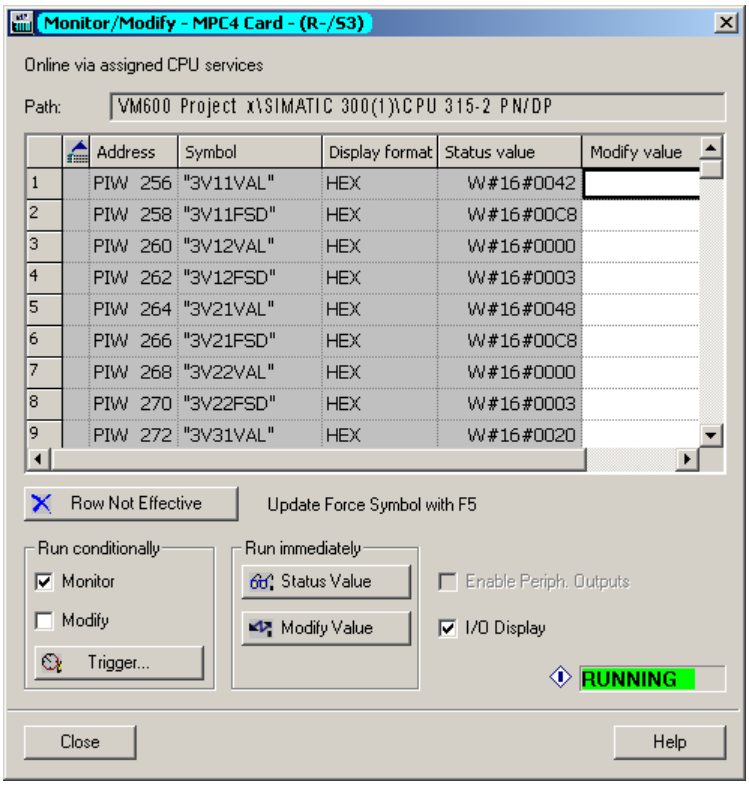



**Figure 10-17:** SIMATIC Manager – la fenêtre « Monitor/Modify »

**4-** Cliquer sur le bouton **Status Value** dans la zone « Run immediately » de la fenêtre « Monitor/Modify » et la fenêtre se met à jour. C'est-à-dire que toutes les valeurs de statut vont s'actualiser en même temps.

**5-** Activer la case à cocher **Monitor** dans la zone « Run conditionally » de la fenêtre « Monitor/Modify » et la fenêtre se met à jour en permanence.

### 10.7.4.3 Affecter des valeurs d'équipements

Cette opération s'exécute en utilisant l'éditeur de symboles SIMATIC, accessible depuis SIMATIC Manager.

**1-** Double-cliquer sur les objets, en commençant par l'icône objet du projet , dans la fenêtre de gauche de SIMATIC Manager, pour pouvoir descendre jusqu'au niveau S7 Program(1) .

(Il est possible également de cliquer sur les icônes dans l'arborescence et de descendre jusqu'au niveau S7 Program(1) .)

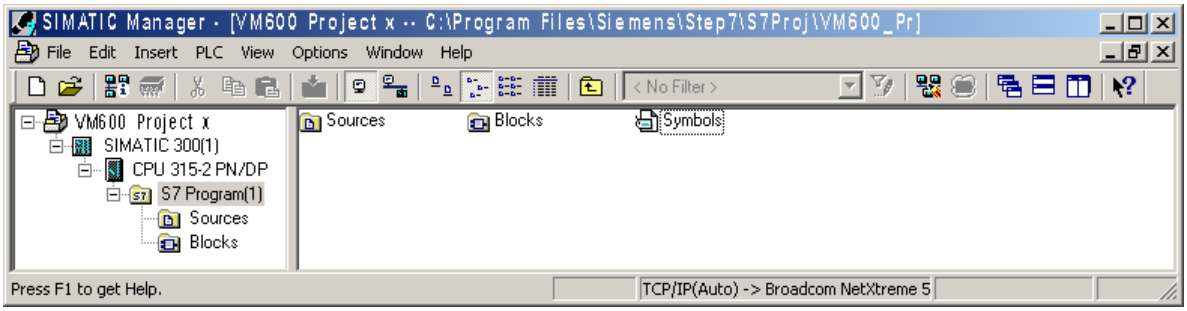



**Figure 10-18:** SIMATIC Manager – accès à l'éditeur de symboles

**2-** Double-cliquer sur l'icône objet « Symbols » dans la fenêtre (principale) de droite de SIMATIC Manager.
La fenêtre « Symbol Editor » apparaît.

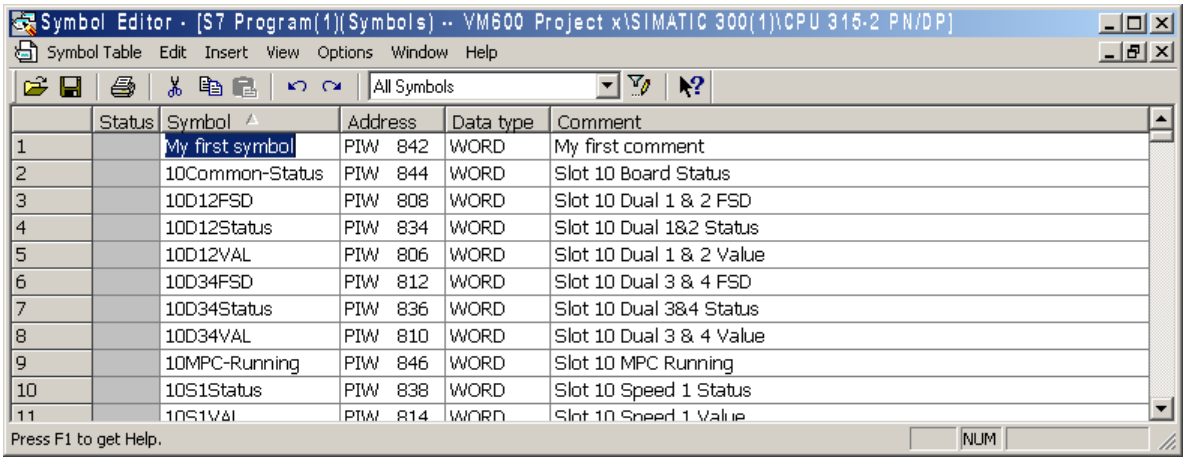



**Figure 10-19:** SIMATIC Manager – accès à la fenêtre « Symbol Editor »

**3-** Cliquer sur la cellule correspondant au symbole ou au commentaire à éditer et saisir la nouvelle information.

**NOTE :** Le tableau de l'Annexe E : Noms des symboles contient une liste des noms de symboles prédéfinis pour les cartes MPC4 et AMC8 pouvant être utilisés ici.

### 10.7.4.4 Créer des tables de variables

Cette opération s'exécute en utilisant l'éditeur de variables SIMATIC, accessible depuis SIMATIC Manager.

**1-** Double-cliquer sur les objets, en commençant par l'icône objet du projet , dans la fenêtre de gauche de SIMATIC Manager, pour pouvoir descendre jusqu'au niveau S7 Program(1) .

(Il est possible également de cliquer sur les icônes ⊞ dans l'arborescence et de descendre jusqu'au niveau S7 Program(1) .)

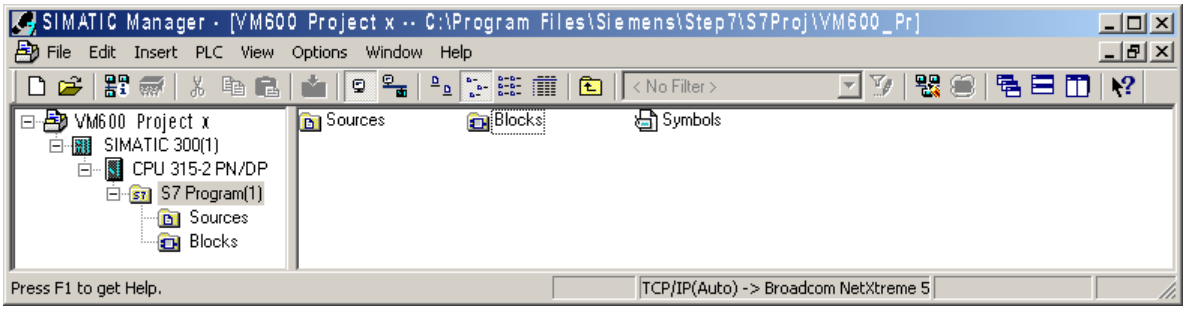



**Figure 10-20:** SIMATIC Manager – accès à l'éditeur de variables

**2-** Faire un clic droit sur l'icône objet « Blocks » dans la fenêtre de gauche, cliquer sur **Insert New Object > Variable Table**.
La fenêtre « Properties - Variable Table » apparaît.

**3-** Entrer un nom pour votre table de variables dans le champ «Symbolic Name» puis cliquer sur **OK.**
La table de variables est ainsi créée.

**4-** Faire un clic droit sur la table de variables à peine créée et sélectionner **Open Object** ou alors double cliquer dessus.
La fenêtre Var apparait.]]

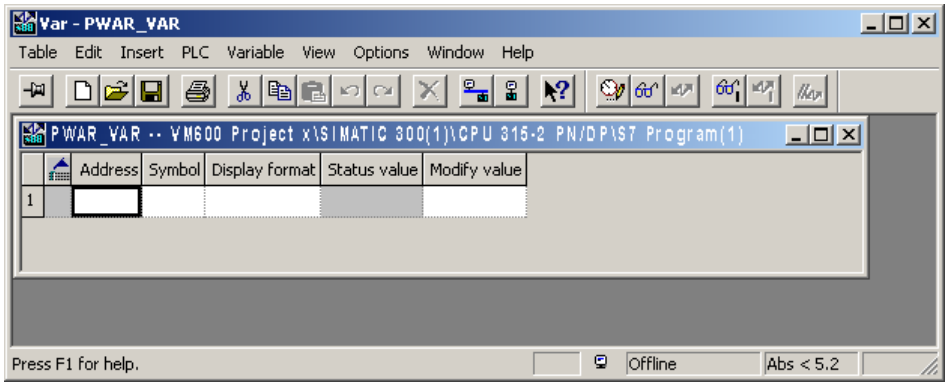



**Figure 10-21:** SIMATIC Manager – la fenêtre « Var »

**5-** Cliquer sur la cellule correspondant à l'adresse ou au symbole à éditer et saisir la nouvelle information.

**NOTE :** Les tables de variables fournissent un moyen rapide pour surveiller aisément les paramètres les plus critiques ou les plus importants d'un projet. (Les tables de variables ne peuvent pas être utilisées pour modifier des paramètres.)

### 10.7.4.5 Vérifier et affecter des noms d'équipements

La communication PROFINET ne fonctionnera pas si le nom d'équipement configuré dans HW Config ne correspond pas au nom actuel de l'équipement. Cependant, il est possible de contrôler le nom attribué aux équipements et de modifier le nom d'un équipement même si celui-ci est déjà en service.

**1-** Dans SIMATIC HW Config, cliquer sur l'objet (1)vm600 pour le sélectionner.

**2-** Cliquer sur **PLC > Ethernet > Verify Device Name**.
La fenêtre « Verify Device Name » apparaît.

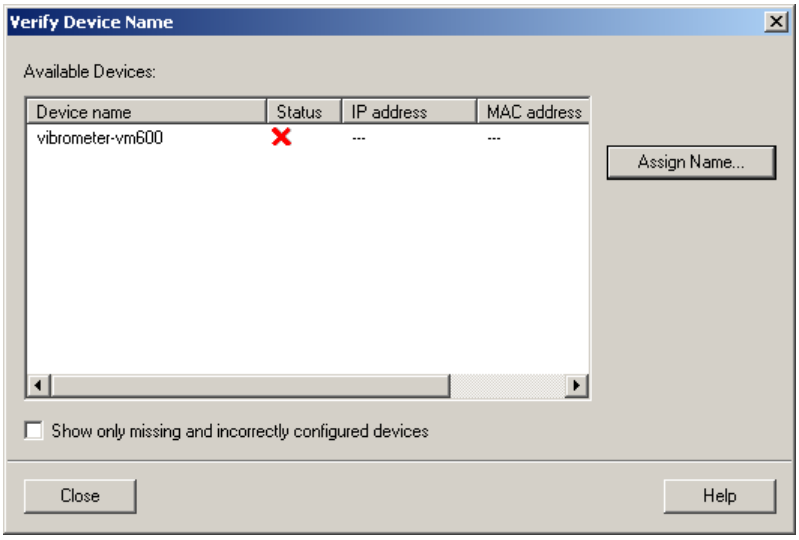



**Figure 10-22:** HW Config – la fenêtre « Verify Device Name »

**3-** Pour modifier le nom d'un équipement déjà en service, cliquer sur l'équipement dans la liste « Available Devices » pour le sélectionner puis cliquer sur le bouton **Assign Name**.
(Il est possible également de cliquer sur **PLC > Ethernet > Assign Device Name**.)
La fenêtre « Assign device name » apparaît.

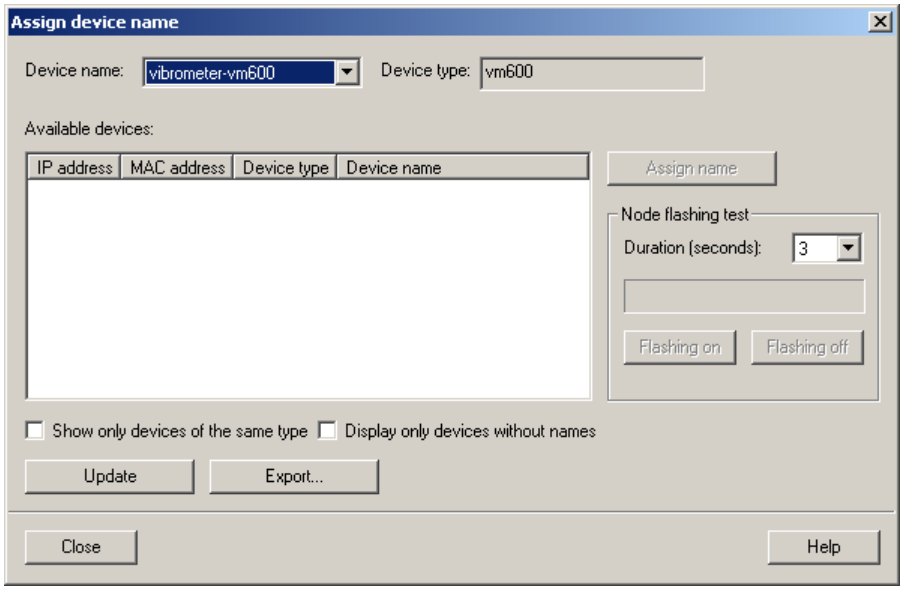



**Figure 10-23:** HW Config – la fenêtre « Assign device name »

**4-** Cliquer sur l'équipement dans la liste « Available devices » pour le sélectionner, saisir le nouveau nom pour l'équipement dans la colonne « Device name » et cliquer sur le bouton **Update**.
Le nom de l'équipement sera actualisé.

# 11 ÉLIMINATION DES PRODUITS EN FIN DE VIE

Un système de surveillance basé sur un rack VM600 est un produit électrique/électronique et doit par conséquent être éliminé de manière acceptable à la fin de sa vie utile. Ceci est important pour réduire la pollution et améliorer l'efficacité des ressources.

**NOTE :** Pour des raisons environnementales et économiques, les équipements électriques et électroniques en fin de vie doivent être collectés et traités séparément des autres déchets : ils ne doivent pas être mis en décharge (dépotoirs, décharges, déchetteries, etc.).

En Europe (Union européenne), les produits électriques/électroniques en fin de vie sont classés comme déchets d'équipements électriques et électroniques (DEEE) et sont soumis aux exigences de la directive 2012/19/UE de l'Union européenne (UE) relative aux déchets équipements électriques et électroniques (communément appelée directive DEEE).

Conformément à la réglementation DEEE, tous les déchets d'équipements électriques et électroniques doivent être collectés séparément, puis traités et éliminés conformément aux meilleures techniques disponibles et respectueuses de l'environnement. En effet, les déchets électroniques (ou e-déchets) peuvent contenir des substances nocives pour l'environnement et/ou pour la santé humaine. En outre, les déchets électroniques constituent également une source précieuse de matières premières pouvant contribuer à une économie circulaire.

Le symbole DEEE (une « poubelle sur roues barrée ») est placé sur l'étiquette du produit pour indiquer que l'équipement doit être correctement traité et éliminé en fin de vie (voir la Figure 11-1).

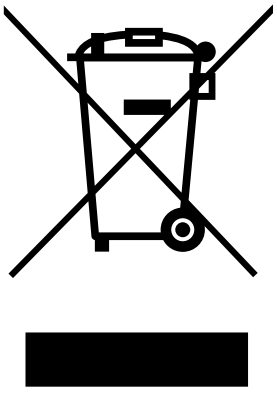


**Figure 11-1:** Symbole DEEE

Bien qu'un certain nombre de pays non membres de l'UE aient adopté des règlements sur les DEEE, différentes lois et réglementations sur l'élimination des produits en fin de vie s'appliquent dans d'autres pays et régions du monde. En conséquence, veuillez consulter les autorités locales pour obtenir les informations et les conseils pertinents pour votre pays et votre région.

**NOTE :** À la fin de sa vie utile, un système de surveillance à base de racks VM600 doit être éliminé de manière écologique.
Dans les États membres de l'Union européenne, la directive DEEE est applicable.
Dans d'autres pays et régions du monde, différentes lois et réglementations peuvent s'appliquer ; veuillez par conséquent consulter les autorités locales.

Pour obtenir des informations complémentaires et des conseils sur l'élimination des produits, contactez votre représentant Meggitt local. Vous pouvez aussi vous adresser à notre bureau principal :

Département environnement, santé et sécurité
Meggitt SA
Route de Moncor 4
Case postale
1701 Fribourg
Suisse

Téléphone : +41 26 407 11 11
Email : ehs@ch.meggitt.com
Site web : www.meggittsensing.com/energy

# 12 ASSISTANCE CLIENT

## 12.1 Pour nous contacter

Le réseau mondial d'assistance client de Meggitt propose toute une série de services comprenant 12.2 Assistance technique et 12.3 Assistance commerciale et réparations. Pour l'assistance client, veuillez contacter votre représentant local Meggitt. Sinon, contactez notre bureau principal :

Service client
Meggitt SA
Route de Moncor 4
Case postale
1701 Fribourg
Switzerland

Téléphone : +41 26 407 11 11
Email : energysupport@ch.meggitt.com
Site web : www.meggittsensing.com/energy

## 12.2 Assistance technique

L'équipe d'assistance technique de Meggitt offre une assistance technique avant et après la vente comprenant :

- Des conseils généraux
- Des conseils techniques
- L'identification des pannes
- Des déplacements sur le terrain.

**NOTE :** Pour de plus amples informations, veuillez contacter Meggitt (voir 12.1 Pour nous contacter).

## 12.3 Assistance commerciale et réparations

L'équipe d'assistance commerciale de Meggitt fournit une assistance avant et après la vente relative aux :

- Nouveaux produits
- Pièces de rechange
- Réparations.

**NOTE :** Si un produit doit être renvoyé à Meggitt pour des réparations, il doit être joint à un Formulaire de retour d'un produit énergétique à remplir, page 12-4.

## 12.4 Commentaires des clients

Dans le cadre de notre engagement permanent à améliorer le service client, votre opinion est la bienvenue. Pour fournir vos commentaires, veuillez remplir le Formulaire de commentaires client d'un produit énergétique à la page 12-7 et le retourner au bureau principal de Meggitt (voir la section 12.1 Pour nous contacter).

# RÉPARATIONS ET RETOURS

## Procédure de retour d'un produit énergétique

Si un produit énergétique Meggitt vibro-meter® doit être renvoyé à Meggitt Suisse, veuillez utiliser la procédure de retour de produit en ligne disponible sur le site web de Meggitt vibro-meter®, à l'adresse:
www.meggittsensing.com/energy/service-and-support/repair

Comme indiqué sur le site web, la procédure de retour d'un produit est la suivante :

**1-** Remplissez et soumettez en ligne le **Formulaire de retour d'un produit énergétique** qui est disponible sur le site web (note : * indique un champ obligatoire).

Pour chaque produit énergétique à renvoyer, un formulaire de retour de produit énergétique distinct doit être rempli et soumis en ligne.
Lorsqu'un formulaire de retour de produit énergétique est soumis en ligne, un e-mail de confirmation incluant un numéro de référence de retour de produit énergétique est envoyé par retour pour confirmer que le formulaire de retour du produit a bien été reçu par Meggitt Suisse.
Veuillez utiliser le numéro de référence de retour du produit énergétique dans toutes les communications futures concernant le retour de votre produit.

**2-** Complétez et incluez un certificat d'utilisateur final.

Pour chaque produit énergétique à renvoyer, un certificat d'utilisateur final associé est également requis.
Le certificat d'utilisateur final à usage unique est recommandé pour les petites organisations qui manipulent peu de produits et le certificat d'utilisateur final annuel est recommandé pour les grandes organisations qui gèrent de nombreux produits.
L'un ou l'autre de ces certificats peut être utilisé pour couvrir plusieurs produits.

**NOTE :** Rendez-vous sur le site web ou contactez notre service clientèle (voir la section 12.1 Pour nous contacter) pour obtenir le formulaire de certificat d'utilisateur final approprié.

**3-** Envoyez le produit énergétique avec des copies imprimées de l'e-mail (ou des e-mails) de confirmation et du ou des certificat(s) de l'utilisateur final à Meggitt Suisse, à l'adresse

**Réparations de produits énergétiques, Meggitt SA, Route de Moncor 4, Case postale, 1701 Fribourg, Suisse**.

Un e-mail de confirmation séparé (copie imprimée) est requis pour chaque produit à renvoyer, bien qu'un seul certificat d'utilisateur final (copie imprimée) puisse être utilisé pour plusieurs produits.

**4-** En outre, un bon de commande d'une valeur de CHF 0,00 doit également être envoyé à Meggitt Suisse, afin de permettre le diagnostic initial du problème.

**NOTE :** Le **Formulaire de retour d'un produit énergétique** reproduit ci-dessous est inclus pour permettre la collecte des informations requises pour la validation et la soumission en ligne.

# Formulaire de retour d'un produit énergétique

## Informations de contact

Prénom :*

Nom de famille :*

Profession :

Société :*

Adresse :*

Pays :*

Email :*

Téléphone :*

Fax :

## Informations sur le produit

Type de produit :*

Référence (PNR) :*

Numéro de série (SER) :

Note : Entrez « Inconnu » si le numéro de série (SER) n'est pas connu.

Produit Ex :

☐ Oui

☐ Non

Produit SIL :*

☐ Oui

☐ Non

Numéro de commande Meggitt SA :

Date d'achat (jj.mm.aaaa) :

Produit sous garantie :

☐ Oui

☐ Non

☐ Ne sait pas

Site d'installation :

Utilisateur final :

## Informations de retour

Motif du retour :*

☐ Réparation

☐ Panne d'origine

Si le motif du retour est « Réparation », veuillez répondre aux questions suivantes :*

Type de panne :

☐ Fonctionnement continu

☐ Fonctionnement intermittent

☐ Dépend de la température

Combien de temps l'appareil a-t-il fonctionné avant la panne ?

Description de la panne :

Veuillez fournir une description détaillée afin de faciliter le diagnostic du problème.

Si le motif du retour est « Panne d'origine », veuillez répondre aux questions suivantes :*

Type de la panne d'origine :

☐ Produit endommagé

☐ Configuration du produit incorrecte

☐ Livraison d'un produit erroné

☐ Problème avec la documentation/l'étiquetage

☐ Produit non opérationnel à l'arrivée

Informations complémentaires :

Veuillez fournir autant d'informations que possible afin de faciliter le diagnostic du problème.

**Informations sur les produits Ex – informations complémentaires requises pour les produits Ex uniquement**

Le produit est-il installé dans une zone dangereuse (atmosphère potentiellement explosive) ? :

☐ Oui

☐ Non

Si le produit est installé dans une zone dangereuse, veuillez répondre aux questions suivantes :

Combien de temps l'appareil a-t-il fonctionné avant la panne ? :

Informations complémentaires :

**Informations sur les produits SIL- informations complémentaires requises pour les produits SIL uniquement***

Note : Pour les produits SIL utilisés dans des contextes/systèmes de sécurité fonctionnelle, cette section d'**informations sur le produit SIL** doit être complétée.

Le produit est-il installé dans un système lié à la sécurité ? :*

☐ Oui

☐ Non

Si le produit est installé dans un système lié à la sécurité, veuillez répondre aux questions suivantes :*

Le système a-t-il échoué** en mode sans échec ? :* (En d'autres termes, le relais de sécurité a fonctionné mais s'est déclenché sans raison.)

☐ Oui

☐ Non

☐ Sans objet

Le système a-t-il échoué ** dans un état dangereux ? :* (Autrement dit, la panne n'a pas abouti à l'état de sécurité.)

☐ Oui

☐ Non

☐ Sans objet

Combien de temps l'appareil a-t-il fonctionné avant la panne (en heures) ? :*

Informations complémentaires :

** Un voyant LED défectueux est considéré comme une défaillance esthétique.

# COMMENTAIRES

## Formulaire de commentaires client d'un produit énergétique

### Informations sur le manuel

Titre du manuel :

*Mise en réseau du VM600*

| | | | |
|---|---|---|---|
| Référence : | MAVM600-NET/F | Version : | Édition 10 |
| Date d'édition : | Février 2021 | | |

### Informations de contact

Prénom :*

Nom de famille :*

Profession :

Société :*

Adresse :*

Pays :*

Email :*

Téléphone :*

Fax :

## Commentaires – général

Veuillez répondre aux questions suivantes :

| | | |
|---|---|---|
| Le document est-il bien structuré ? | ☐ Yes | ☐ No |
| L'information est-elle exacte du point de vue technique ? | ☐ Yes | ☐ No |
| Avez-vous besoin d'autres détails techniques ? | ☐ Yes | ☐ No |
| Les instructions sont-elles claires et complètes ? | ☐ Yes | ☐ No |
| Les descriptions sont-elles faciles à comprendre ? | ☐ Yes | ☐ No |
| Les exemples et diagrammes / photographies sont-ils d'une grande aide ? | ☐ Yes | ☐ No |
| Y-a-t-il assez d'exemples et diagrammes / photographies ? | ☐ Yes | ☐ No |
| Le style / les termes utilisés sont-il faciles à lire ? | ☐ Yes | ☐ No |
| Existe-il des informations qui font défaut ? | ☐ Yes | ☐ No |

Veuillez inclure toute information supplémentaire dans la section "Commentaires – supplémentaires" ci-dessous.

## Commentaires – supplémentaires

Informations complémentaires :

Veuillez fournir autant de commentaires que possible afin de nous aider à améliorer notre documentation produit.
Continuer sur une feuille séparée si nécessaire …

# ANNEXE A : DÉFINITIONS DE REGISTRE MODBUS MPC4

Les définitions de registre de la carte MPC4 pour Modbus sont disponibles dans les tableaux de mappage d'adresses suivants :

- Tableau A-1: Lecture de bits ou de registres d'entrées discrètes (codes de fonction Modbus 01 et 02)
- Tableau A-2: Lecture de registres d'entrées ou de maintien (codes de fonction Modbus 03 et 04).

Ces tableaux contiennent une colonne (CPUM < 71) qui indique si le registre Modbus est disponible dans une version particulière du logiciel CPUM. En général, la plupart des registres Modbus sont disponibles, mais les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) ont accès à tous les registres Modbus.

Ces tableaux contiennent les informations requises pour calculer les adresses de démarrage Modbus (MSA, Modbus Starting Addresses). Ces adresses doivent être connues pour accéder aux registres requis et peuvent être déterminées à l'aide de la technique suivante :

**1-** Décidez dans quel logement la carte sera insérée (Snum, Slot Number).

**2-** Utilisez le tableau approprié (Tableau A-1 ou Tableau A-2) pour définir le décalage du registre d'adresse (Aoff, Address Offset). Le choix du tableau dépend du code de fonction Modbus à utiliser.

**3-** Utilisez les formules indiquées dans la colonne Adresse de démarrage Modbus (MSA) du tableau pour calculer l'adresse MSA.

---

**NOTE :** Pour obtenir le numéro de registre, il vous suffit d'ajouter 1 à la valeur MSA calculée.

---

Exemple :

Votre carte MPC4 est dans le logement 5 et vous voulez lire le résultat des combinaisons logiques d'alarmes pour les huit fonctions de base que vous avez définies dans le logiciel MPS (elles correspondent aux bits b0 à b7).

Pour ce faire, vous devez utiliser la fonction Modbus 04 - Lecture de registres d'entrées.

En consultant le Tableau A-2, vous constatez que Aoff = 1144.

Finalement, en utilisant l'équation fournie dans le tableau, vous obtenez l'adresse Modbus de démarrage :

MSA = ( ( logement − 3 ) * 3 ) + Aoff = ( 5 − 3 ) * 3 ) + 1144 = 1150.

Par conséquent, le numéro de registre = 1150 + 1 = 1151.

---

**NOTE :** Une liste complète des adresses Modbus de démarrage présentée sous forme d'un fichier Excel® est disponible sur demande auprès de Meggitt.

---

**Tableau A-1 :** Définitions de registre de la MPC4
Lecture de bits internes ou de registres d'entrées discrètes (codes de fonction Modbus 01 et 02)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|
| De 3 à 14 | 0 | Canal 1, sortie 1 | Valeur | Le point est défini | ( ( logement – 3 ) * 96 ) + Aoff | Bit | Même comportement |
| | 1 | | | Alarme+ | | | |
| | 2 | | | Alarme– | | | |
| | 3 | | | Danger+ | | | |
| | 4 | | | Danger– | | | |
| | 5 | | | Capteur OK | | | |
| | 8 | Canal 1, sortie 2 | | Le point est défini | | | |
| | 9 | | | Alarme+ | | | |
| | 10 | | | Alarme– | | | |
| | 11 | | | Danger+ | | | |
| | 12 | | | Danger– | | | |
| | 13 | | | Capteur OK | | | |
| | 16 | Canal 2, sortie 1 | | Le point est défini | | | |
| | 17 | | | Alarme+ | | | |
| | 18 | | | Alarme– | | | |
| | 19 | | | Danger+ | | | |
| | 20 | | | Danger– | | | |
| | 21 | | | Capteur OK | | | |
| | 24 | Canal 2, sortie 2 | | Le point est défini | | | |
| | 25 | | | Alarme+ | | | |
| | 26 | | | Alarme– | | | |
| | 27 | | | Danger+ | | | |
| | 28 | | | Danger– | | | |
| | 29 | | | Capteur OK | | | |
| | 32 | Canal 3, sortie 1 | | Le point est défini | | | |
| | 33 | | | Alarme+ | | | |
| | 34 | | | Alarme– | | | |
| | 35 | | | Danger+ | | | |
| | 36 | | | Danger– | | | |
| | 37 | | | Capteur OK | | | |
| | 40 | Canal 3, sortie 2 | | Le point est défini | | | |
| | 41 | | | Alarme+ | | | |
| | 42 | | | Alarme– | | | |
| | 43 | | | Danger+ | | | |
| | 44 | | | Danger– | | | |
| | 45 | | | Capteur OK | | | |

**Tableau A-1 :** Définitions de registre de la MPC4
Lecture de bits internes ou de registres d'entrées discrètes (codes de fonction Modbus 01 et 02)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|
| De 3 à 14 | 48 | Canal 4, sortie 1 | Valeur | Le point est défini | ( ( logement – 3 ) * 96 ) + Aoff | Bit | Même comportement |
| | 49 | | | Alarme+ | | | |
| | 50 | | | Alarme– | | | |
| | 51 | | | Danger+ | | | |
| | 52 | | | Danger– | | | |
| | 53 | | | Capteur OK | | | |
| | 56 | Canal 4, sortie 2 | | Le point est défini | | | |
| | 57 | | | Alarme+ | | | |
| | 58 | | | Alarme– | | | |
| | 59 | | | Danger+ | | | |
| | 60 | | | Danger– | | | |
| | 61 | | | Capteur OK | | | |
| | 64 | Double canal 1 et 2 | | Le point est défini | | | |
| | 65 | | | Alarme+ | | | |
| | 66 | | | Alarme– | | | |
| | 67 | | | Danger+ | | | |
| | 68 | | | Danger– | | | |
| | 69 | | | Capteur OK | | | |
| | 72 | Double canal 3 et 4 | | Le point est défini | | | |
| | 73 | | | Alarme+ | | | |
| | 74 | | | Alarme– | | | |
| | 75 | | | Danger+ | | | |
| | 76 | | | Danger– | | | |
| | 77 | | | Capteur OK | | | |
| | 80 | Vitesse 1 | | Le point est défini | | | |
| | 81 | | | Alarme+ | | | |
| | 82 | | | Alarme– | | | |
| | 85 | | | Capteur OK | | | |
| | 88 | Vitesse 2 | | Le point est défini | | | |
| | 89 | | | Alarme+ | | | |
| | 90 | | | Alarme– | | | |
| | 93 | | | Capteur OK | | | |

**Tableau A-2 :** Définitions de registre de la MPC4
Lecture de registres d'entrées ou de maintien (codes de fonction Modbus 03 et 04)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Canal | Type de valeur interne | Description de la valeur | Adresse de démarrage Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|
| De 3 à 14 | 0 | Canal 1, sortie 1 | Valeur | VAL | ( ( logement – 3 ) * 22 ) + Aoff | Non signé | Même comportement |
| | 1 | | Configuration | FSD | | | |
| | 2 | Canal 1, sortie 2 | Valeur | VAL | | | |
| | 3 | | Configuration | FSD | | | |
| | 4 | Canal 2, sortie 1 | Valeur | VAL | | | |
| | 5 | | Configuration | FSD | | | |
| | 6 | Canal 2, sortie 2 | Valeur | VAL | | | |
| | 7 | | Configuration | FSD | | | |
| | 8 | Canal 3, sortie 1 | Valeur | VAL | | | |
| | 9 | | Configuration | FSD | | | |
| | 10 | Canal 3, sortie 2 | Valeur | VAL | | | |
| | 11 | | Configuration | FSD | | | |
| | 12 | Canal 4, sortie 1 | Valeur | VAL | | | |
| | 13 | | Configuration | FSD | | | |
| | 14 | Canal 4, sortie 2 | Valeur | VAL | | | |
| | 15 | | Configuration | FSD | | | |
| | 16 | Double canal 1 et 2, sortie 1 | Valeur | VAL | | | |
| | 17 | | Configuration | FSD | | | |
| | 18 | Double canal 3 et 4, sortie 1 | Valeur | VAL | | | |
| | 19 | | Configuration | FSD | | | |
| | 20 | Vitesse 1 | Valeur | VAL | | | |
| | 21 | Vitesse 2 | Valeur | FSD | | | |
| | 1000 | Canal 1, sortie 1 | Valeur | État alarme (voir les Tableau 8-6 et Tableau 9-10) | ( ( logement – 3 ) * 12 ) + Aoff | | Seuls les bits b0 à b6 sont disponibles |
| | 1001 | Canal 1, sortie 2 | | | | | |
| | 1002 | Canal 2, sortie 1 | | | | | |
| | 1003 | Canal 2, sortie 2 | | | | | |
| | 1004 | Canal 3, sortie 1 | | | | | |
| | 1005 | Canal 3, sortie 2 | | | | | |
| | 1006 | Canal 4, sortie 1 | | | | | |
| | 1007 | Canal 4, sortie 2 | | | | | |
| | 1008 | Double canal 1 et 2 | | | | | |
| | 1009 | Double canal 3 et 4 | | | | | |
| | 1010 | Vitesse 1 | | | | | |
| | 1011 | Vitesse 2 | | | | | |
| | 1144 | Fonction de base et avancée | | Résultat logique | ( ( logement – 3 ) * 3) + Aoff | | Non disponible |
| | 1145 | État de carte n°1 | | État de carte (voir les Tableau 9-12, Tableau 9-13 et Tableau 9-14) | | | |
| | 1146 | État de carte n°2 | | | | | |

# ANNEXE B : DÉFINITIONS DE REGISTRE MODBUS AMC8

Les définitions de registre de la carte AMC8 pour Modbus sont disponibles dans les tableaux de mappage d'adresses suivants :

- Tableau B-1: Lecture de bits ou de registres d'entrées discrètes (code de fonction Modbus 01)
- Tableau B-2: Lecture de registres d'entrées discrètes (code de fonction Modbus 02)
- Tableau B-3: Lecture de registres d'entrées (code de fonction Modbus 03)
- Tableau B-4: Lecture de registres de maintien (code de fonction Modbus 04).

Ces tableaux contiennent une colonne (CPUM < 71) qui indique si le registre Modbus est disponible dans une version particulière du logiciel CPUM. En général, la plupart des registres Modbus sont disponibles, mais les cartes CPUM exécutant la version logicielle 071 (ou ultérieure) ont accès à tous les registres Modbus.

Ces tableaux contiennent les informations requises pour calculer les adresses Modbus de départ (MSA, Modbus Starting Addresses). Ces adresses doivent être connues pour accéder aux registres requis et peuvent être déterminées à l'aide de la technique suivante :

**1-** Décidez dans quel logement la carte sera insérée (Snum, Slot Number).

**2-** Utilisez le tableau approprié (Tableau B-1, Tableau B-2, Tableau B-3 ou Tableau B-4) pour définir le décalage du registre d'adresse (Aoff, Address Offset). Le choix du tableau dépend du code de fonction Modbus à utiliser.

**3-** Utilisez les formules suivantes pour calculer l'adresse Modbus de départ (MSA) :

$$\mathbf{MSA = 4096 + Aoff + (256 * Snum)} \qquad \text{(B.1)}$$

**NOTE :** Pour obtenir le numéro de registre, il vous suffit d'ajouter 1 à la valeur MSA calculée.

Exemple :

Votre carte AMC8 est dans le logement 14 et vous voulez lire le résultat des combinaisons logiques d'alarmes pour les 16 fonctions de base que vous avez définies dans le logiciel MPS (elles correspondent aux bits b0 à b15).

Pour ce faire, vous devez utiliser la fonction Modbus 04 - Lecture de registres d'entrées.

En consultant le Tableau B-4, vous constatez que Aoff = 24.

Finalement, en utilisant l'équation (B.1), vous obtenez l'adresse Modbus de départ:

MSA = 4096 + 24 + (256 * 14) = 7704.

Par conséquent, le numéro de registre = 7704 + 1 = 7705.

**NOTE :** Une liste complète des adresses Modbus de départ présentée sous forme d'un fichier Excel® est disponible sur demande auprès de Meggitt.

**Tableau B-1 :** Définitions de registre d'AMC8
Lecture de bits ou de registres d'entrées discrètes (code de fonction Modbus 01)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 0 | S/O | Canal 1 | Configuration | Alerte– Bas Activé | Voir l'équation (B.1) | Bit | Même comportement |
| | 1 | | Canal 2 | | | | | |
| | 2 | | Canal 3 | | | | | |
| | 3 | | Canal 4 | | | | | |
| | 4 | | Canal 5 | | | | | |
| | 5 | | Canal 6 | | | | | |
| | 6 | | Canal 7 | | | | | |
| | 7 | | Canal 8 | | | | | |
| | 8 | | Multi-canal 1 | | | | | |
| | 9 | | Multi-canal 2 | | | | | |
| | 10 | | Multi-canal 3 | | | | | |
| | 11 | | Multi-canal 4 | | | | | |
| | 12 | | Canal 1 | | Alerte+ Haut Activé | | | |
| | 13 | | Canal 2 | | | | | |
| | 14 | | Canal 3 | | | | | |
| | 15 | | Canal 4 | | | | | |
| | 16 | | Canal 5 | | | | | |
| | 17 | | Canal 6 | | | | | |
| | 18 | | Canal 7 | | | | | |
| | 19 | | Canal 8 | | | | | |
| | 20 | | Multi-canal 1 | | | | | |
| | 21 | | Multi-canal 2 | | | | | |
| | 22 | | Multi-canal 3 | | | | | |
| | 23 | | Multi-canal 4 | | | | | |
| | 24 | | Canal 1 | | Danger– Bas Activé | | | |
| | 25 | | Canal 2 | | | | | |
| | 26 | | Canal 3 | | | | | |
| | 27 | | Canal 4 | | | | | |
| | 28 | | Canal 5 | | | | | |
| | 29 | | Canal 6 | | | | | |
| | 30 | | Canal 7 | | | | | |
| | 31 | | Canal 8 | | | | | |
| | 32 | | Multi-canal 1 | | | | | |
| | 33 | | Multi-canal 2 | | | | | |
| | 34 | | Multi-canal 3 | | | | | |
| | 35 | | Multi-canal 4 | | | | | |

**Tableau B-1 :** Définitions de registre d'AMC8
Lecture de bits ou de registres d'entrées discrètes (code de fonction Modbus 01)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 36 | S/O | Canal 1 | Configuration | Danger+ Haut Activé | Voir l'équation (B.1) (B.1) | Bit | Même comportement |
| | 37 | | Canal 2 | | | | | |
| | 38 | | Canal 3 | | | | | |
| | 39 | | Canal 4 | | | | | |
| | 40 | | Canal 5 | | | | | |
| | 41 | | Canal 6 | | | | | |
| | 42 | | Canal 7 | | | | | |
| | 43 | | Canal 8 | | | | | |
| | 44 | | Multi-canal 1 | | | | | |
| | 45 | | Multi-canal 2 | | | | | |
| | 46 | | Multi-canal 3 | | | | | |
| | 47 | | Multi-canal 4 | | | | | |

**Tableau B-2 :** Définitions de registre de AMC8
Lecture de registres d'entrées discrètes (code de fonction Modbus 02)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 0 | S/O | Canal 1 | Valeur | Alarme+ | Voir l'équation (B.1) (B.1) | Bit | Même comportement |
| | 1 | | Canal 2 | | | | | |
| | 2 | | Canal 3 | | | | | |
| | 3 | | Canal 4 | | | | | |
| | 4 | | Canal 5 | | | | | |
| | 5 | | Canal 6 | | | | | |
| | 6 | | Canal 7 | | | | | |
| | 7 | | Canal 8 | | | | | |
| | 8 | | Multi-canal 1 | | | | | |
| | 9 | | Multi-canal 2 | | | | | |
| | 10 | | Multi-canal 3 | | | | | |
| | 11 | | Multi-canal 4 | | | | | |
| | 12 | | Canal 1 | | Alarme– | | | |
| | 13 | | Canal 2 | | | | | |
| | 14 | | Canal 3 | | | | | |
| | 15 | | Canal 4 | | | | | |
| | 16 | | Canal 5 | | | | | |
| | 17 | | Canal 6 | | | | | |
| | 18 | | Canal 7 | | | | | |
| | 19 | | Canal 8 | | | | | |
| | 20 | | Multi-canal 1 | | | | | |
| | 21 | | Multi-canal 2 | | | | | |
| | 22 | | Multi-canal 3 | | | | | |
| | 23 | | Multi-canal 4 | | | | | |
| | 24 | | Canal 1 | | Danger+ | | | |
| | 25 | | Canal 2 | | | | | |
| | 26 | | Canal 3 | | | | | |
| | 27 | | Canal 4 | | | | | |
| | 28 | | Canal 5 | | | | | |
| | 29 | | Canal 6 | | | | | |
| | 30 | | Canal 7 | | | | | |
| | 31 | | Canal 8 | | | | | |
| | 32 | | Multi-canal 1 | | | | | |
| | 33 | | Multi-canal 2 | | | | | |
| | 34 | | Multi-canal 3 | | | | | |
| | 35 | | Multi-canal 4 | | | | | |

**Tableau B-2 :** Définitions de registre de AMC8
Lecture de registres d'entrées discrètes (code de fonction Modbus 02)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 36 | S/O | Canal 1 | Valeur | Danger– | Voir l'équation (B.1) (B.1) | Bit | Même comportement |
| | 37 | | Canal 2 | | | | | |
| | 38 | | Canal 3 | | | | | |
| | 39 | | Canal 4 | | | | | |
| | 40 | | Canal 5 | | | | | |
| | 41 | | Canal 6 | | | | | |
| | 42 | | Canal 7 | | | | | |
| | 43 | | Canal 8 | | | | | |
| | 44 | | Multi-canal 1 | | | | | |
| | 45 | | Multi-canal 2 | | | | | |
| | 46 | | Multi-canal 3 | | | | | |
| | 47 | | Multi-canal 4 | | | | | |
| | 48 | | Canal 1 | | Échec du OK global du canal | | | |
| | 49 | | Canal 2 | | | | | |
| | 50 | | Canal 3 | | | | | |
| | 51 | | Canal 4 | | | | | |
| | 52 | | Canal 5 | | | | | |
| | 53 | | Canal 6 | | | | | |
| | 54 | | Canal 7 | | | | | |
| | 55 | | Canal 8 | | | | | |
| | 56 | | Multi-canal 1 | | | | | |
| | 57 | | Multi-canal 2 | | | | | |
| | 58 | | Multi-canal 3 | | | | | |
| | 59 | | Multi-canal 4 | | | | | |
| | 60 | | Canal 1 | | Erreur ADC | | | |
| | 61 | | Canal 2 | | | | | |
| | 62 | | Canal 3 | | | | | |
| | 63 | | Canal 4 | | | | | |
| | 64 | | Canal 5 | | | | | |
| | 65 | | Canal 6 | | | | | |
| | 66 | | Canal 7 | | | | | |
| | 67 | | Canal 8 | | | | | |
| | 68 | | Multi-canal 1 | | | | | |
| | 69 | | Multi-canal 2 | | | | | |
| | 70 | | Multi-canal 3 | | | | | |
| | 71 | | Multi-canal 4 | | | | | |
| | 72 | | Canal 1 | | Pause ADC | | | |
| | 73 | | Canal 2 | | | | | |
| | 74 | | Canal 3 | | | | | |

**Tableau B-2 :** Définitions de registre de AMC8
Lecture de registres d'entrées discrètes (code de fonction Modbus 02)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 75 | S/O | Canal 4 | Valeur | Pause ADC | Voir l'équation (B.1) (B.1) | Bit | Même comportement |
| | 76 | | Canal 5 | | | | | |
| | 77 | | Canal 6 | | | | | |
| | 78 | | Canal 7 | | | | | |
| | 79 | | Canal 8 | | | | | |
| | 80 | | Multi-canal 1 | | | | | |
| | 81 | | Multi-canal 2 | | | | | |
| | 82 | | Multi-canal 3 | | | | | |
| | 83 | | Multi-canal 4 | | | | | |
| | 84 | | Canal 1 | | Erreur de verrouillage PLL ADC | | | |
| | 85 | | Canal 2 | | | | | |
| | 86 | | Canal 3 | | | | | |
| | 87 | | Canal 4 | | | | | |
| | 88 | | Canal 5 | | | | | |
| | 89 | | Canal 6 | | | | | |
| | 90 | | Canal 7 | | | | | |
| | 91 | | Canal 8 | | | | | |
| | 92 | | Multi-canal 1 | | | | | |
| | 93 | | Multi-canal 2 | | | | | |
| | 94 | | Multi-canal 3 | | | | | |
| | 95 | | Multi-canal 4 | | | | | |
| | 96 | | Canal 1 | | Erreur de transmission ADC | | | |
| | 97 | | Canal 2 | | | | | |
| | 98 | | Canal 3 | | | | | |
| | 99 | | Canal 4 | | | | | |
| | 100 | | Canal 5 | | | | | |
| | 101 | | Canal 6 | | | | | |
| | 102 | | Canal 7 | | | | | |
| | 103 | | Canal 8 | | | | | |
| | 104 | | Multi-canal 1 | | | | | |
| | 105 | | Multi-canal 2 | | | | | |
| | 106 | | Multi-canal 3 | | | | | |
| | 107 | | Multi-canal 4 | | | | | |
| | 108 | | Canal 1 | | Erreur de configuration dynamique | | | |
| | 109 | | Canal 2 | | | | | |
| | 110 | | Canal 3 | | | | | |
| | 111 | | Canal 4 | | | | | |
| | 112 | | Canal 5 | | | | | |
| | 113 | | Canal 6 | | | | | |

**Tableau B-2 :** Définitions de registre de AMC8
Lecture de registres d'entrées discrètes (code de fonction Modbus 02)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 114 | S/O | Canal 7 | Valeur | Erreur de configuration dynamique | Voir l'équation (B.1) | Bit | Même comportement |
| | 115 | | Canal 8 | | | | | |
| | 116 | | Multi-canal 1 | | | | | |
| | 117 | | Multi-canal 2 | | | | | |
| | 118 | | Multi-canal 3 | | | | | |
| | 119 | | Multi-canal 4 | | | | | |
| | 120 | | Canal 1 | | Échec du résultat BIT | | | |
| | 121 | | Canal 2 | | | | | |
| | 122 | | Canal 3 | | | | | |
| | 123 | | Canal 4 | | | | | |
| | 124 | | Canal 5 | | | | | |
| | 125 | | Canal 6 | | | | | |
| | 126 | | Canal 7 | | | | | |
| | 127 | | Canal 8 | | | | | |
| | 128 | | Multi-canal 1 | | | | | |
| | 129 | | Multi-canal 2 | | | | | |
| | 130 | | Multi-canal 3 | | | | | |
| | 131 | | Multi-canal 4 | | | | | |
| | 132 | | Canal 1 | | Pas d'échantillonnage | | | |
| | 133 | | Canal 2 | | | | | |
| | 134 | | Canal 3 | | | | | |
| | 135 | | Canal 4 | | | | | |
| | 136 | | Canal 5 | | | | | |
| | 137 | | Canal 6 | | | | | |
| | 138 | | Canal 7 | | | | | |
| | 139 | | Canal 8 | | | | | |
| | 140 | | Multi-canal 1 | | | | | |
| | 141 | | Multi-canal 2 | | | | | |
| | 142 | | Multi-canal 3 | | | | | |
| | 143 | | Multi-canal 4 | | | | | |
| | 144 | | Canal 1 | | Échec erreur OK | | | |
| | 145 | | Canal 2 | | | | | |
| | 146 | | Canal 3 | | | | | |
| | 147 | | Canal 4 | | | | | |
| | 148 | | Canal 5 | | | | | |
| | 149 | | Canal 6 | | | | | |
| | 150 | | Canal 7 | | | | | |
| | 151 | | Canal 8 | | | | | |
| | 152 | | Multi-canal 1 | | | | | |

**Tableau B-2 :** Définitions de registre de AMC8
Lecture de registres d'entrées discrètes (code de fonction Modbus 02)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 153 | S/O | Multi-canal 2 | Valeur | Échec erreur OK | Voir l'équation (B.1) (B.1)(B.1) | Bit | Même comportement |
| | 154 | | Multi-canal 3 | | | | | |
| | 155 | | Multi-canal 4 | | | | | |
| | 156 | | Canal 1 | | Erreur de linéarisation | | | |
| | 157 | | Canal 2 | | | | | |
| | 158 | | Canal 3 | | | | | |
| | 159 | | Canal 4 | | | | | |
| | 160 | | Canal 5 | | | | | |
| | 161 | | Canal 6 | | | | | |
| | 162 | | Canal 7 | | | | | |
| | 163 | | Canal 8 | | | | | |
| | 164 | | Multi-canal 1 | | | | | |
| | 165 | | Multi-canal 2 | | | | | |
| | 166 | | Multi-canal 3 | | | | | |
| | 167 | | Multi-canal 4 | | | | | |
| | 168 | | Canal 1 | | Erreur de soudure froide | | | |
| | 169 | | Canal 2 | | | | | |
| | 170 | | Canal 3 | | | | | |
| | 171 | | Canal 4 | | | | | |
| | 172 | | Canal 5 | | | | | |
| | 173 | | Canal 6 | | | | | |
| | 174 | | Canal 7 | | | | | |
| | 175 | | Canal 8 | | | | | |
| | 176 | | Multi-canal 1 | | | | | |
| | 177 | | Multi-canal 2 | | | | | |
| | 178 | | Multi-canal 3 | | | | | |
| | 179 | | Multi-canal 4 | | | | | |
| | 180 | | S/O | | Fonction de base 1 | | | |
| | 181 | | | | Fonction de base 2 | | | |
| | 182 | | | | Fonction de base 3 | | | |
| | 183 | | | | Fonction de base 4 | | | |
| | 184 | | | | Fonction de base 5 | | | |
| | 185 | | | | Fonction de base 6 | | | |
| | 186 | | | | Fonction de base 7 | | | |
| | 187 | | | | Fonction de base 8 | | | |
| | 188 | | | | Fonction de base 9 | | | |
| | 189 | | | | Fonction de base 10 | | | |
| | 190 | | | | Fonction de base 11 | | | |
| | 191 | | | | Fonction de base 12 | | | |

**Tableau B-2 :** Définitions de registre de AMC8
Lecture de registres d'entrées discrètes (code de fonction Modbus 02)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 192 | S/O | S/O | Valeur | Fonction de base 13 | Voir l'équation (B.1) (B.1) | Bit | Même comportement |
| | 193 | | | | Fonction de base 14 | | | |
| | 194 | | | | Fonction de base 15 | | | |
| | 195 | | | | Fonction de base 16 | | | |
| | 196 | | | | Fonction avancée 1 | | | |
| | 197 | | | | Fonction avancée 2 | | | |
| | 198 | | | | Fonction avancée 3 | | | |
| | 199 | | | | Fonction avancée 4 | | | |
| | 200 | | | | Fonction avancée 5 | | | |
| | 201 | | | | Fonction avancée 6 | | | |
| | 202 | | | | Fonction avancée 7 | | | |
| | 203 | | | | Fonction avancée 8 | | | |

**Tableau B-3 :** Définitions de registre de l'AMC8
Lecture de registres d'entrées (code de fonction Modbus 03)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 0 | 0-15 | Canal 1 | Valeur | Valeurs traitées (voir le Tableau 8-7) | Voir l'équation (B.1) (B.1) | Float32 | Même comportement |
| | 1 | 16-31 | | | | | | |
| | 2 | 0-15 | Canal 2 | | | | | |
| | 3 | 16-31 | | | | | | |
| | 4 | 0-15 | Canal 3 | | | | | |
| | 5 | 16-31 | | | | | | |
| | 6 | 0-15 | Canal 4 | | | | | |
| | 7 | 16-31 | | | | | | |
| | 8 | 0-15 | Canal 5 | | | | | |
| | 9 | 16-31 | | | | | | |
| | 10 | 0-15 | Canal 6 | | | | | |
| | 11 | 16-31 | | | | | | |
| | 12 | 0-15 | Canal 7 | | | | | |
| | 13 | 16-31 | | | | | | |
| | 14 | 0-15 | Canal 8 | | | | | |
| | 15 | 16-31 | | | | | | |
| | 16 | 0-15 | Multi-canal 1 | | | | | |
| | 17 | 16-31 | | | | | | |
| | 18 | 0-15 | Multi-canal 2 | | | | | |
| | 19 | 16-31 | | | | | | |
| | 20 | 0-15 | Multi-canal 3 | | | | | |
| | 21 | 16-31 | | | | | | |
| | 22 | 0-15 | Multi-canal 4 | | | | | |
| | 23 | 16-31 | | | | | | |
| | 24 | 0-15 | S/O | | Résultats logiques (voir le Tableau 8-7) | | U32 | |
| | 25 | 16-31 | | | | | | |
| | 26 | 0-15 | Canal 1 | | État alarme (voir le Tableau 8-7) | | U16 | |
| | 27 | | Canal 2 | | | | | |
| | 28 | | Canal 3 | | | | | |
| | 29 | | Canal 4 | | | | | |
| | 30 | | Canal 5 | | | | | |
| | 31 | | Canal 6 | | | | | |
| | 32 | | Canal 7 | | | | | |

**Tableau B-3 :** Définitions de registre de l'AMC8
Lecture de registres d'entrées (code de fonction Modbus 03)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 33 | 0-15 | Canal 8 | Valeur | État alarme (voir le Tableau 8-7) | Voir l'équation (B.1) (B.1) | U16 | Même comporte- |
| | 34 | | Multi-canal 1 | | | | | |
| | 35 | | Multi-canal 2 | | | | | |
| | 36 | | Multi-canal 3 | | | | | |
| | 37 | | Multi-canal 4 | | | | | |
| De 3 à 14 | 38 | 0-15 | Canal 1 | Valeur | Alarme utilisée (voir le Tableau 9-15) | | U16 | Non disponible |
| | 39 | | Canal 2 | | | | | |
| | 40 | | Canal 3 | | | | | |
| | 41 | | Canal 4 | | | | | |
| | 42 | | Canal 5 | | | | | |
| | 43 | | Canal 6 | | | | | |
| | 44 | | Canal 7 | | | | | |
| | 45 | | Canal 8 | | | | | |
| | 46 | | Multi-canal 1 | | | | | |
| | 47 | | Multi-canal 2 | | | | | |
| | 48 | | Multi-canal 3 | | | | | |
| | 49 | | Multi-canal 4 | | | | | |
| | 50 | | S/O | | État de commande (Tableau 9-16) | | | |

**Tableau B-4 :** Définitions de registre d'AMC8
Lecture de registres de maintien (code de fonction Modbus 04)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 0 | 0-15 | Canal 1 | Configuration | Valeur minimum affichée | Voir l'équation (B.1) (B.1) | Float32 | Même comportement |
| | 1 | 16-31 | | | | | | |
| | 2 | 0-15 | Canal 2 | | | | | |
| | 3 | 16-31 | | | | | | |
| | 4 | 0-15 | Canal 3 | | | | | |
| | 5 | 16-31 | | | | | | |
| | 6 | 0-15 | Canal 4 | | | | | |
| | 7 | 16-31 | | | | | | |
| | 8 | 0-15 | Canal 5 | | | | | |
| | 9 | 16-31 | | | | | | |
| | 10 | 0-15 | Canal 6 | | | | | |
| | 11 | 16-31 | | | | | | |
| | 12 | 0-15 | Canal 7 | | | | | |
| | 13 | 16-31 | | | | | | |
| | 14 | 0-15 | Canal 8 | | | | | |
| | 15 | 16-31 | | | | | | |
| | 16 | 0-15 | Multi-canal 1 | | | | | |
| | 17 | 16-31 | | | | | | |
| | 18 | 0-15 | Multi-canal 2 | | | | | |
| | 19 | 16-31 | | | | | | |
| | 20 | 0-15 | Multi-canal 3 | | | | | |
| | 21 | 16-31 | | | | | | |
| | 22 | 0-15 | Multi-canal 4 | | | | | |
| | 23 | 16-31 | | | | | | |
| | 24 | 0-15 | Canal 1 | | Valeur maximum affichée | | | |
| | 25 | 16-31 | | | | | | |
| | 26 | 0-15 | Canal 2 | | | | | |
| | 27 | 16-31 | | | | | | |
| | 28 | 0-15 | Canal 3 | | | | | |
| | 29 | 16-31 | | | | | | |
| | 30 | 0-15 | Canal 4 | | | | | |
| | 31 | 16-31 | | | | | | |
| | 32 | 0-15 | Canal 5 | | | | | |
| | 33 | 16-31 | | | | | | |
| | 34 | 0-15 | Canal 6 | | | | | |
| | 35 | 16-31 | | | | | | |

**Tableau B-4 :** Définitions de registre d'AMC8
Lecture de registres de maintien (code de fonction Modbus 04)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 36 | 0-15 | Canal 7 | Configuration | Valeur maximum affichée | Voir l'équation (B.1) (B.1) | Float32 | Même comportement |
| | 37 | 16-31 | | | | | | |
| | 38 | 0-15 | Canal 8 | | | | | |
| | 39 | 16-31 | | | | | | |
| | 40 | 0-15 | Multi-canal 1 | | | | | |
| | 41 | 16-31 | | | | | | |
| | 42 | 0-15 | Multi-canal 2 | | | | | |
| | 43 | 16-31 | | | | | | |
| | 44 | 0-15 | Multi-canal 3 | | | | | |
| | 45 | 16-31 | | | | | | |
| | 46 | 0-15 | Multi-canal 4 | | | | | |
| | 47 | 16-31 | | | | | | |
| | 48 | 0-15 | Canal 1 | | Alerte– Bas | | | |
| | 49 | 16-31 | | | | | | |
| | 50 | 0-15 | Canal 2 | | | | | |
| | 51 | 16-31 | | | | | | |
| | 52 | 0-15 | Canal 3 | | | | | |
| | 53 | 16-31 | | | | | | |
| | 54 | 0-15 | Canal 4 | | | | | |
| | 55 | 16-31 | | | | | | |
| | 56 | 0-15 | Canal 5 | | | | | |
| | 57 | 16-31 | | | | | | |
| | 58 | 0-15 | Canal 6 | | | | | |
| | 59 | 16-31 | | | | | | |
| | 60 | 0-15 | Canal 7 | | | | | |
| | 61 | 16-31 | | | | | | |
| | 62 | 0-15 | Canal 8 | | | | | |
| | 63 | 16-31 | | | | | | |
| | 64 | 0-15 | Multi-canal 1 | | | | | |
| | 65 | 16-31 | | | | | | |
| | 66 | 0-15 | Multi-canal 2 | | | | | |
| | 67 | 16-31 | | | | | | |
| | 68 | 0-15 | Multi-canal 3 | | | | | |
| | 69 | 16-31 | | | | | | |
| | 70 | 0-15 | Multi-canal 4 | | | | | |
| | 71 | 16-31 | | | | | | |
| | 72 | 0-15 | Canal 1 | | Alerte+ Haut | | | |
| | 73 | 16-31 | | | | | | |

**Tableau B-4 :** Définitions de registre d'AMC8
Lecture de registres de maintien (code de fonction Modbus 04)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 74 | 0-15 | Canal 2 | Configuration | Alerte+ Haut | Voir l'équation (B.1) (B.1) | Float32 | Même comportement |
| | 75 | 16-31 | | | | | | |
| | 76 | 0-15 | Canal 3 | | | | | |
| | 77 | 16-31 | | | | | | |
| | 78 | 0-15 | Canal 4 | | | | | |
| | 79 | 16-31 | | | | | | |
| | 80 | 0-15 | Canal 5 | | | | | |
| | 81 | 16-31 | | | | | | |
| | 82 | 0-15 | Canal 6 | | | | | |
| | 83 | 16-31 | | | | | | |
| | 84 | 0-15 | Canal 7 | | | | | |
| | 85 | 16-31 | | | | | | |
| | 86 | 0-15 | Canal 8 | | | | | |
| | 87 | 16-31 | | | | | | |
| | 88 | 0-15 | Multi-canal 1 | | | | | |
| | 89 | 16-31 | | | | | | |
| | 90 | 0-15 | Multi-canal 2 | | | | | |
| | 91 | 16-31 | | | | | | |
| | 92 | 0-15 | Multi-canal 3 | | | | | |
| | 93 | 16-31 | | | | | | |
| | 94 | 0-15 | Multi-canal 4 | | | | | |
| | 95 | 16-31 | | | | | | |
| | 96 | 0-15 | Canal 1 | | Danger– Faible | | | |
| | 97 | 16-31 | | | | | | |
| | 98 | 0-15 | Canal 2 | | | | | |
| | 99 | 16-31 | | | | | | |
| | 100 | 0-15 | Canal 3 | | | | | |
| | 101 | 16-31 | | | | | | |
| | 102 | 0-15 | Canal 4 | | | | | |
| | 103 | 16-31 | | | | | | |
| | 104 | 0-15 | Canal 5 | | | | | |
| | 105 | 16-31 | | | | | | |
| | 106 | 0-15 | Canal 6 | | | | | |
| | 107 | 16-31 | | | | | | |
| | 108 | 0-15 | Canal 7 | | | | | |
| | 109 | 16-31 | | | | | | |
| | 110 | 0-15 | Canal 8 | | | | | |
| | 111 | 16-31 | | | | | | |

**Tableau B-4 :** Définitions de registre d'AMC8
Lecture de registres de maintien (code de fonction Modbus 04)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 112 | 0-15 | Multi-canal 1 | Configuration | Danger– Bas | Voir l'équation (B.1) (B.1) | Float32 | Même comportement |
| | 113 | 16-31 | | | | | | |
| | 114 | 0-15 | Multi-canal 2 | | | | | |
| | 115 | 16-31 | | | | | | |
| | 116 | 0-15 | Multi-canal 3 | | | | | |
| | 117 | 16-31 | | | | | | |
| | 118 | 0-15 | Multi-canal 4 | | | | | |
| | 119 | 16-31 | | | | | | |
| | 120 | 0-15 | Canal 1 | | Danger+ Haut | | | |
| | 121 | 16-31 | | | | | | |
| | 122 | 0-15 | Canal 2 | | | | | |
| | 123 | 16-31 | | | | | | |
| | 124 | 0-15 | Canal 3 | | | | | |
| | 125 | 16-31 | | | | | | |
| | 126 | 0-15 | Canal 4 | | | | | |
| | 127 | 16-31 | | | | | | |
| | 128 | 0-15 | Canal 5 | | | | | |
| | 129 | 16-31 | | | | | | |
| | 130 | 0-15 | Canal 6 | | | | | |
| | 131 | 16-31 | | | | | | |
| | 132 | 0-15 | Canal 7 | | | | | |
| | 133 | 16-31 | | | | | | |
| | 134 | 0-15 | Canal 8 | | | | | |
| | 135 | 16-31 | | | | | | |
| | 136 | 0-15 | Multi-canal 1 | | | | | |
| | 137 | 16-31 | | | | | | |
| | 138 | 0-15 | Multi-canal 2 | | | | | |
| | 139 | 16-31 | | | | | | |
| | 140 | 0-15 | Multi-canal 3 | | | | | |
| | 141 | 16-31 | | | | | | |
| | 142 | 0-15 | Multi-canal 4 | | | | | |
| | 143 | 16-31 | | | | | | |

**Tableau B-4 :** Définitions de registre d'AMC8
Lecture de registres de maintien (code de fonction Modbus 04)

| Numéro de logement du rack (Snum) | Décalage du registre d'adresse (Aoff) | Bit(s) | Canal | Type de valeur interne | Description de la valeur | Adresse de départ Modbus (MSA) | Type de valeur | CPUM < 71 |
|---|---|---|---|---|---|---|---|---|
| De 3 à 14 | 144 | 0-15 | Canal 1 | Configuration | Unité de sortie | Voir l'équation (B.1) (B.1) | U16 | Même comportement |
| | 145 | 0-15 | Canal 2 | | | | | |
| | 146 | 0-15 | Canal 3 | | | | | |
| | 147 | 0-15 | Canal 4 | | | | | |
| | 148 | 0-15 | Canal 5 | | | | | |
| | 149 | 0-15 | Canal 6 | | | | | |
| | 150 | 0-15 | Canal 7 | | | | | |
| | 151 | 0-15 | Canal 8 | | | | | |
| | 152 | 0-15 | Multi-canal 1 | | | | | |
| | 153 | 0-15 | Multi-canal 2 | | | | | |
| | 154 | 0-15 | Multi-canal 3 | | | | | |
| | 155 | 0-15 | Multi-canal 4 | | | | | |

# ANNEXE C : DÉFINITION DES LOGEMENTS PROFINET MPC4

Les définitions des logements des cartes MPC4 pour PROFINET sont illustrées à l'aide de l'extrait suivant d'un fichier de configuration `modbusDefault.cfg` type pour une carte MPC4 dans le logement 3 d'un rack VM600.

```
...

[GLOBAL]
...
IS_PROFINET_ACTIVED = YES

[MAPPING]

////////////////////////////////////////////////
///////////////// Slot3 ////////////////////////
////////////////////////////////////////////////

///////////////// Alarm and Status /////////////

//  Slot3, Channel 1, Output 1
$MPC4S3C1O1Used = Slot3:MPC4:Config:C1:O1:Used
$MPC4S3C1O1Apos = Slot3:MPC4:C1:O1:A+
$MPC4S3C1O1Aneg = Slot3:MPC4:C1:O1:A-
$MPC4S3C1O1Dpos = Slot3:MPC4:C1:O1:D+
$MPC4S3C1O1Dneg = Slot3:MPC4:C1:O1:D-
$MPC4S3C1O1SOK  = Slot3:MPC4:Status:C1:SOK
$MPC4S3C1O1UsedAndApos = AND($MPC4S3C1O1Used, $MPC4S3C1O1Apos)
$MPC4S3C1O1UsedAndAneg = AND($MPC4S3C1O1Used, $MPC4S3C1O1Aneg)
$MPC4S3C1O1UsedAndDpos = AND($MPC4S3C1O1Used, $MPC4S3C1O1Dpos)
$MPC4S3C1O1UsedAndDneg = AND($MPC4S3C1O1Used, $MPC4S3C1O1Dneg)
$MPC4S3C1O1UsedAndSOK  = AND($MPC4S3C1O1Used, $MPC4S3C1O1SOK)

//  Slot3, Channel 1, Output 2
$MPC4S3C1O2Used = Slot3:MPC4:Config:C1:O2:Used
$MPC4S3C1O2Apos = Slot3:MPC4:C1:O2:A+
$MPC4S3C1O2Aneg = Slot3:MPC4:C1:O2:A-
$MPC4S3C1O2Dpos = Slot3:MPC4:C1:O2:D+
$MPC4S3C1O2Dneg = Slot3:MPC4:C1:O2:D-
$MPC4S3C1O2SOK  = Slot3:MPC4:Status:C1:SOK
$MPC4S3C1O2UsedAndApos = AND($MPC4S3C1O2Used, $MPC4S3C1O2Apos)
$MPC4S3C1O2UsedAndAneg = AND($MPC4S3C1O2Used, $MPC4S3C1O2Aneg)
$MPC4S3C1O2UsedAndDpos = AND($MPC4S3C1O2Used, $MPC4S3C1O2Dpos)
$MPC4S3C1O2UsedAndDneg = AND($MPC4S3C1O2Used, $MPC4S3C1O2Dneg)
$MPC4S3C1O2UsedAndSOK  = AND($MPC4S3C1O2Used, $MPC4S3C1O2SOK)
```

```
//  Slot3, Channel 2, Output 1
$MPC4S3C2O1Used = Slot3:MPC4:Config:C2:O1:Used
$MPC4S3C2O1Apos = Slot3:MPC4:C2:O1:A+
$MPC4S3C2O1Aneg = Slot3:MPC4:C2:O1:A-
$MPC4S3C2O1Dpos = Slot3:MPC4:C2:O1:D+
$MPC4S3C2O1Dneg = Slot3:MPC4:C2:O1:D-
$MPC4S3C2O1SOK  = Slot3:MPC4:Status:C2:SOK
$MPC4S3C2O1UsedAndApos = AND($MPC4S3C2O1Used, $MPC4S3C2O1Apos)
$MPC4S3C2O1UsedAndAneg = AND($MPC4S3C2O1Used, $MPC4S3C2O1Aneg)
$MPC4S3C2O1UsedAndDpos = AND($MPC4S3C2O1Used, $MPC4S3C2O1Dpos)
$MPC4S3C2O1UsedAndDneg = AND($MPC4S3C2O1Used, $MPC4S3C2O1Dneg)
$MPC4S3C2O1UsedAndSOK  = AND($MPC4S3C2O1Used, $MPC4S3C2O1SOK)

//  Slot3, Channel 2, Output 2
$MPC4S3C2O2Used = Slot3:MPC4:Config:C2:O2:Used
$MPC4S3C2O2Apos = Slot3:MPC4:C2:O2:A+
$MPC4S3C2O2Aneg = Slot3:MPC4:C2:O2:A-
$MPC4S3C2O2Dpos = Slot3:MPC4:C2:O2:D+
$MPC4S3C2O2Dneg = Slot3:MPC4:C2:O2:D-
$MPC4S3C2O2SOK  = Slot3:MPC4:Status:C2:SOK
$MPC4S3C2O2UsedAndApos = AND($MPC4S3C2O2Used, $MPC4S3C2O2Apos)
$MPC4S3C2O2UsedAndAneg = AND($MPC4S3C2O2Used, $MPC4S3C2O2Aneg)
$MPC4S3C2O2UsedAndDpos = AND($MPC4S3C2O2Used, $MPC4S3C2O2Dpos)
$MPC4S3C2O2UsedAndDneg = AND($MPC4S3C2O2Used, $MPC4S3C2O2Dneg)
$MPC4S3C2O2UsedAndSOK  = AND($MPC4S3C2O2Used, $MPC4S3C2O2SOK)

//  Slot3, Channel 3, Output 1
$MPC4S3C3O1Used = Slot3:MPC4:Config:C3:O1:Used
$MPC4S3C3O1Apos = Slot3:MPC4:C3:O1:A+
$MPC4S3C3O1Aneg = Slot3:MPC4:C3:O1:A-
$MPC4S3C3O1Dpos = Slot3:MPC4:C3:O1:D+
$MPC4S3C3O1Dneg = Slot3:MPC4:C3:O1:D-
$MPC4S3C3O1SOK  = Slot3:MPC4:Status:C3:SOK
$MPC4S3C3O1UsedAndApos = AND($MPC4S3C3O1Used, $MPC4S3C3O1Apos)
$MPC4S3C3O1UsedAndAneg = AND($MPC4S3C3O1Used, $MPC4S3C3O1Aneg)
$MPC4S3C3O1UsedAndDpos = AND($MPC4S3C3O1Used, $MPC4S3C3O1Dpos)
$MPC4S3C3O1UsedAndDneg = AND($MPC4S3C3O1Used, $MPC4S3C3O1Dneg)
$MPC4S3C3O1UsedAndSOK  = AND($MPC4S3C3O1Used, $MPC4S3C3O1SOK)

//  Slot3, Channel 3, Output 2
$MPC4S3C3O2Used = Slot3:MPC4:Config:C3:O2:Used
$MPC4S3C3O2Apos = Slot3:MPC4:C3:O2:A+
$MPC4S3C3O2Aneg = Slot3:MPC4:C3:O2:A-
$MPC4S3C3O2Dpos = Slot3:MPC4:C3:O2:D+
$MPC4S3C3O2Dneg = Slot3:MPC4:C3:O2:D-
$MPC4S3C3O2SOK  = Slot3:MPC4:Status:C3:SOK
```

```
$MPC4S3C3O2UsedAndApos = AND($MPC4S3C3O2Used, $MPC4S3C3O2Apos)
$MPC4S3C3O2UsedAndAneg = AND($MPC4S3C3O2Used, $MPC4S3C3O2Aneg)
$MPC4S3C3O2UsedAndDpos = AND($MPC4S3C3O2Used, $MPC4S3C3O2Dpos)
$MPC4S3C3O2UsedAndDneg = AND($MPC4S3C3O2Used, $MPC4S3C3O2Dneg)
$MPC4S3C3O2UsedAndSOK  = AND($MPC4S3C3O2Used, $MPC4S3C3O2SOK)

//  Slot3, Channel 4, Output 1
$MPC4S3C4O1Used = Slot3:MPC4:Config:C4:O1:Used
$MPC4S3C4O1Apos = Slot3:MPC4:C4:O1:A+
$MPC4S3C4O1Aneg = Slot3:MPC4:C4:O1:A-
$MPC4S3C4O1Dpos = Slot3:MPC4:C4:O1:D+
$MPC4S3C4O1Dneg = Slot3:MPC4:C4:O1:D-
$MPC4S3C4O1SOK  = Slot3:MPC4:Status:C4:SOK
$MPC4S3C4O1UsedAndApos = AND($MPC4S3C4O1Used, $MPC4S3C4O1Apos)
$MPC4S3C4O1UsedAndAneg = AND($MPC4S3C4O1Used, $MPC4S3C4O1Aneg)
$MPC4S3C4O1UsedAndDpos = AND($MPC4S3C4O1Used, $MPC4S3C4O1Dpos)
$MPC4S3C4O1UsedAndDneg = AND($MPC4S3C4O1Used, $MPC4S3C4O1Dneg)
$MPC4S3C4O1UsedAndSOK  = AND($MPC4S3C4O1Used, $MPC4S3C4O1SOK)

//  Slot3, Channel 4, Output 2
$MPC4S3C4O2Used = Slot3:MPC4:Config:C4:O2:Used
$MPC4S3C4O2Apos = Slot3:MPC4:C4:O2:A+
$MPC4S3C4O2Aneg = Slot3:MPC4:C4:O2:A-
$MPC4S3C4O2Dpos = Slot3:MPC4:C4:O2:D+
$MPC4S3C4O2Dneg = Slot3:MPC4:C4:O2:D-
$MPC4S3C4O2SOK  = Slot3:MPC4:Status:C4:SOK
$MPC4S3C4O2UsedAndApos = AND($MPC4S3C4O2Used, $MPC4S3C4O2Apos)
$MPC4S3C4O2UsedAndAneg = AND($MPC4S3C4O2Used, $MPC4S3C4O2Aneg)
$MPC4S3C4O2UsedAndDpos = AND($MPC4S3C4O2Used, $MPC4S3C4O2Dpos)
$MPC4S3C4O2UsedAndDneg = AND($MPC4S3C4O2Used, $MPC4S3C4O2Dneg)
$MPC4S3C4O2UsedAndSOK  = AND($MPC4S3C4O2Used, $MPC4S3C4O2SOK)

//  Slot3, Multi-Channel 1
$MPC4S3M1Used = Slot3:MPC4:Config:M1:Used
$MPC4S3M1Apos = Slot3:MPC4:M1:A+
$MPC4S3M1Aneg = Slot3:MPC4:M1:A-
$MPC4S3M1Dpos = Slot3:MPC4:M1:D+
$MPC4S3M1Dneg = Slot3:MPC4:M1:D-
$MPC4S3M1SOK  = Slot3:MPC4:Status:M1:SOK
$MPC4S3M1UsedAndApos = AND($MPC4S3M1Used, $MPC4S3M1Apos)
$MPC4S3M1UsedAndAneg = AND($MPC4S3M1Used, $MPC4S3M1Aneg)
$MPC4S3M1UsedAndDpos = AND($MPC4S3M1Used, $MPC4S3M1Dpos)
$MPC4S3M1UsedAndDneg = AND($MPC4S3M1Used, $MPC4S3M1Dneg)
$MPC4S3M1UsedAndSOK  = AND($MPC4S3M1Used, $MPC4S3M1SOK)
```

```
//  Slot3, Multi-Channel 2
$MPC4S3M2Used = Slot3:MPC4:Config:M2:Used
$MPC4S3M2Apos = Slot3:MPC4:M2:A+
$MPC4S3M2Aneg = Slot3:MPC4:M2:A-
$MPC4S3M2Dpos = Slot3:MPC4:M2:D+
$MPC4S3M2Dneg = Slot3:MPC4:M2:D-
$MPC4S3M2SOK  = Slot3:MPC4:Status:M1:SOK
$MPC4S3M2UsedAndApos = AND($MPC4S3M2Used, $MPC4S3M2Apos)
$MPC4S3M2UsedAndAneg = AND($MPC4S3M2Used, $MPC4S3M2Aneg)
$MPC4S3M2UsedAndDpos = AND($MPC4S3M2Used, $MPC4S3M2Dpos)
$MPC4S3M2UsedAndDneg = AND($MPC4S3M2Used, $MPC4S3M2Dneg)
$MPC4S3M2UsedAndSOK  = AND($MPC4S3M2Used, $MPC4S3M2SOK)

//  Slot3, Speed 1
$MPC4S3S1Used = Slot3:MPC4:Config:S1:Used
$MPC4S3S1Apos = Slot3:MPC4:S1:A+
$MPC4S3S1Aneg = Slot3:MPC4:S1:A-
$MPC4S3S1SOK  = Slot3:MPC4:S1:SOK
$MPC4S3S1UsedAndApos = AND($MPC4S3S1Used, $MPC4S3S1Apos)
$MPC4S3S1UsedAndAneg = AND($MPC4S3S1Used, $MPC4S3S1Aneg)
$MPC4S3S1UsedAndSOK  = AND($MPC4S3S1Used, $MPC4S3S1SOK)

//  Slot3, Speed 2
$MPC4S3S2Used = Slot3:MPC4:Config:S2:Used
$MPC4S3S2Apos = Slot3:MPC4:S2:A+
$MPC4S3S2Aneg = Slot3:MPC4:S2:A-
$MPC4S3S2SOK  = Slot3:MPC4:S2:SOK
$MPC4S3S2UsedAndApos = AND($MPC4S3S2Used, $MPC4S3S2Apos)
$MPC4S3S2UsedAndAneg = AND($MPC4S3S2Used, $MPC4S3S2Aneg)
$MPC4S3S2UsedAndSOK  = AND($MPC4S3S2Used, $MPC4S3S2SOK)

//  Slot3, Basic and Advanced Functions
$MPC4S3BF1 = Slot3:MPC4:Logical:BF1
$MPC4S3BF2 = Slot3:MPC4:Logical:BF2
$MPC4S3BF3 = Slot3:MPC4:Logical:BF3
$MPC4S3BF4 = Slot3:MPC4:Logical:BF4
$MPC4S3BF5 = Slot3:MPC4:Logical:BF5
$MPC4S3BF6 = Slot3:MPC4:Logical:BF6
$MPC4S3BF7 = Slot3:MPC4:Logical:BF7
$MPC4S3BF8 = Slot3:MPC4:Logical:BF8
$MPC4S3AF1 = Slot3:MPC4:Logical:AF1
$MPC4S3AF2 = Slot3:MPC4:Logical:AF2
$MPC4S3AF3 = Slot3:MPC4:Logical:AF3
$MPC4S3AF4 = Slot3:MPC4:Logical:AF4
```

```
//  Slot3, Common Monitoring Functions
$MPC4S3MCR  = Slot3:MPC4:CommonStatus:MCR
$MPC4S3CMF  = Slot3:MPC4:CommonStatus:CMF
$MPC4S3COF  = Slot3:MPC4:CommonStatus:COF
$MPC4S3CPE  = Slot3:MPC4:CommonStatus:CPE
$MPC4S3CIE  = Slot3:MPC4:CommonStatus:CIE
$MPC4S3CA   = Slot3:MPC4:CommonStatus:CA
$MPC4S3CD   = Slot3:MPC4:CommonStatus:CD
$MPC4S3CSOL = Slot3:MPC4:CommonStatus:CSOL
$MPC4S3CTL  = Slot3:MPC4:CommonStatus:CTL
$MPC4S3CTOR = Slot3:MPC4:CommonStatus:CTOR
$MPC4S3CDSE = Slot3:MPC4:CommonStatus:CDSE
$MPC4S3CISE = Slot3:MPC4:CommonStatus:CISE
$MPC4S3CCME = Slot3:MPC4:CommonStatus:CCME
$MPC4S3SL   = Slot3:MPC4:CommonStatus:SL
$MPC4S3TM   = Slot3:MPC4:CommonStatus:TM
$MPC4S3DBP  = Slot3:MPC4:CommonStatus:DBP
$MPC4S3AR   = Slot3:MPC4:CommonStatus:AR


00000:3:U = Slot3:MPC4:C1:O1:V
00001:3:U = Slot3:MPC4:C1:O1:FSD
00002:3:U = Slot3:MPC4:C1:O2:V
00003:3:U = Slot3:MPC4:C1:O2:FSD
00004:3:U = Slot3:MPC4:C2:O1:V
00005:3:U = Slot3:MPC4:C2:O1:FSD
00006:3:U = Slot3:MPC4:C2:O2:V
00007:3:U = Slot3:MPC4:C2:O2:FSD
00008:3:U = Slot3:MPC4:C3:O1:V
00009:3:U = Slot3:MPC4:C3:O1:FSD
00010:3:U = Slot3:MPC4:C3:O2:V
00011:3:U = Slot3:MPC4:C3:O2:FSD
00012:3:U = Slot3:MPC4:C4:O1:V
00013:3:U = Slot3:MPC4:C4:O1:FSD
00014:3:U = Slot3:MPC4:C4:O2:V
00015:3:U = Slot3:MPC4:C4:O2:FSD
00016:3:U = Slot3:MPC4:M1:V
00017:3:U = Slot3:MPC4:M1:FSD
00018:3:U = Slot3:MPC4:M2:V
00019:3:U = Slot3:MPC4:M2:FSD
00020:3:U = Slot3:MPC4:S1:SPEED
00021:3:U = Slot3:MPC4:S2:SPEED
00022:3:U = Pack($MPC4S3C1O1Used,$MPC4S3C1O1UsedAndApos,
$MPC4S3C1O1UsedAndAneg, $MPC4S3C1O1UsedAndDpos,$MPC4S3C1O1UsedAndDneg,
$MPC4S3C1O1UsedAndSOK,0,0,0,0,0,0,0,0,0,0)
```

```
00023:3:U = Pack($MPC4S3C1O2Used,$MPC4S3C1O2UsedAndApos,
$MPC4S3C1O2UsedAndAneg,$MPC4S3C1O2UsedAndDpos,$MPC4S3C1O2UsedAndDneg,
$MPC4S3C1O2UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00024:3:U = Pack($MPC4S3C2O1Used,$MPC4S3C2O1UsedAndApos,
$MPC4S3C2O1UsedAndAneg,$MPC4S3C2O1UsedAndDpos,$MPC4S3C2O1UsedAndDneg,
$MPC4S3C2O1UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00025:3:U = Pack($MPC4S3C2O2Used,$MPC4S3C2O2UsedAndApos,
$MPC4S3C2O2UsedAndAneg,$MPC4S3C2O2UsedAndDpos,$MPC4S3C2O2UsedAndDneg,
$MPC4S3C2O2UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00026:3:U = Pack($MPC4S3C3O1Used,$MPC4S3C3O1UsedAndApos,
$MPC4S3C3O1UsedAndAneg,$MPC4S3C3O1UsedAndDpos,$MPC4S3C3O1UsedAndDneg,
$MPC4S3C3O1UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00027:3:U = Pack($MPC4S3C3O2Used,$MPC4S3C3O2UsedAndApos,
$MPC4S3C3O2UsedAndAneg,$MPC4S3C3O2UsedAndDpos,$MPC4S3C3O2UsedAndDneg,
$MPC4S3C3O2UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00028:3:U = Pack($MPC4S3C4O1Used,$MPC4S3C4O1UsedAndApos,
$MPC4S3C4O1UsedAndAneg,$MPC4S3C4O1UsedAndDpos,$MPC4S3C4O1UsedAndDneg,
$MPC4S3C4O1UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00029:3:U = Pack($MPC4S3C4O2Used,$MPC4S3C4O2UsedAndApos,
$MPC4S3C4O2UsedAndAneg,$MPC4S3C4O2UsedAndDpos,$MPC4S3C4O2UsedAndDneg,
$MPC4S3C4O2UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00030:3:U = Pack($MPC4S3M1Used,$MPC4S3M1UsedAndApos,
$MPC4S3M1UsedAndAneg,$MPC4S3M1UsedAndDpos,$MPC4S3M1UsedAndDneg,
$MPC4S3M1UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00031:3:U = Pack($MPC4S3M2Used,$MPC4S3M2UsedAndApos,
$MPC4S3M2UsedAndAneg,$MPC4S3M2UsedAndDpos,$MPC4S3M2UsedAndDneg,
$MPC4S3M2UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00032:3:U = Pack($MPC4S3S1Used,$MPC4S3S1UsedAndApos,
$MPC4S3S1UsedAndAneg,0,0,$MPC4S3S1UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00033:3:U = Pack($MPC4S3S2Used,$MPC4S3S2UsedAndApos,
$MPC4S3S2UsedAndAneg,0,0,$MPC4S3S2UsedAndSOK,0,0,0,0,0,0,0,0,0,0)

00034:3:U = PACK($MPC4S3BF1,$MPC4S3BF2,$MPC4S3BF3,$MPC4S3BF4,
$MPC4S3BF5,$MPC4S3BF6,$MPC4S3BF7,$MPC4S3BF8,$MPC4S3AF1,$MPC4S3AF2,
$MPC4S3AF3,$MPC4S3AF4,0,0,0,0)

00035:3:U = PACK($MPC4S3CMF,$MPC4S3COF,$MPC4S3CPE,$MPC4S3CIE,
$MPC4S3CA,$MPC4S3CD,$MPC4S3CSOL,$MPC4S3CTL,$MPC4S3CTOR,$MPC4S3CDSE,
$MPC4S3CISE,$MPC4S3CCME,$MPC4S3SL,$MPC4S3TM,$MPC4S3DBP,$MPC4S3AR)

00036:3:U = PACK($MPC4S3MCR,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)


...


PROFINET_CREATE SLOT3 0 36
```

Comme indiqué ci-dessus, dans l'extrait pour une carte MPC4 dans le logement 3 :

- PROFINET est activé dans la section `[GLOBAL]` au moyen de la balise `IS_PROFINET_ACTIVED`.
- Les sorties MPC4 sont définies et organisées dans la section `[MAPPING]`, sur une base logement par logement (carte), au moyen des balises `$MPC4...`.

**NOTE :** Les balises `$MPC4...` peuvent utiliser des opérateurs logiques pour combiner des indicateurs d'état (bits).

- Le mappage du registre Modbus MPC4 est défini dans la section `[MAPPING]`, sur une base logement par logement (carte), au moyen de la syntaxe de mappage de registre Modbus.

  Exemple : `00000:3:U = Slot3:MPC4:C1:O1:V`
  Voir la section 9.2.4 La section MAPPING pour plus d'informations.

**NOTE :** Le mappage du registre Modbus peut utiliser la commande pack (`Pack(...)`) pour combiner des balises `$MPC4...`.

- Les logements PROFINET sont définis dans la section `[MAPPING]`, à la fin du fichier de configuration, au moyen de balises `PROFINET_CREATE`.

  Exemple : `PROFINET_CREATE SLOT3 0 36`, où
  - `SLOT3` est l'identifiant du logement PROFINET.
    Le numéro d'identification du logement PROFINET peut correspondre au numéro de logement du rack VM600 ou un schéma de numérotation différent peut être utilisé.
  - `0` est la première adresse de registre Modbus mappée sur ce logement PROFINET.
  - `36` est la dernière adresse de registre Modbus mappée sur ce logement PROFINET.
    Les première et dernière adresses de registre Modbus mappées sur un logement PROFINET doivent contenir un bloc de données contigu.

Les structures de données sous-jacentes du logement PROFINET sont définies dans un fichier GSDML (voir la section 10.5 Fichiers GSDML). Le mappage du registre Modbus MPC4 et les définitions des logements PROFINET configurées dans la section `[MAPPING]` du fichier de configuration doivent correspondre aux structures de données définies dans le fichier GSDML.

CETTE PAGE EST LAISSÉE INTENTIONNELLEMENT VIDE

# ANNEXE D : DÉFINITIONS DES LOGEMENTS PROFINET AMC8

Les définitions des logements des cartes AMC8 pour PROFINET sont illustrées à l'aide de l'extrait suivant d'un fichier de configuration `modbusDefault.cfg` type pour une carte AMC8 dans le logement 7 d'un rack VM600.

```
...

[GLOBAL]
...
IS_PROFINET_ACTIVED = YES

[MAPPING]

////////////////////////////////////////////////
///////////////// Slot7 ////////////////////////
////////////////////////////////////////////////

///////////////// Alarm and Status /////////////

//  Slot 7, Alarm+
$S7C1APos = Slot7:AMC8:C1:A+
$S7C2APos = Slot7:AMC8:C2:A+
$S7C3APos = Slot7:AMC8:C3:A+
$S7C4APos = Slot7:AMC8:C4:A+
$S7C5APos = Slot7:AMC8:C5:A+
$S7C6APos = Slot7:AMC8:C6:A+
$S7C7APos = Slot7:AMC8:C7:A+
$S7C8APos = Slot7:AMC8:C8:A+
$S7M1APos = Slot7:AMC8:M1:A+
$S7M2APos = Slot7:AMC8:M2:A+
$S7M3APos = Slot7:AMC8:M3:A+
$S7M4APos = Slot7:AMC8:M4:A+

//  Slot 7, Alarm-
$S7C1ANeg = Slot7:AMC8:C1:A-
$S7C2ANeg = Slot7:AMC8:C2:A-
$S7C3ANeg = Slot7:AMC8:C3:A-
$S7C4ANeg = Slot7:AMC8:C4:A-
$S7C5ANeg = Slot7:AMC8:C5:A-
$S7C6ANeg = Slot7:AMC8:C6:A-
$S7C7ANeg = Slot7:AMC8:C7:A-
$S7C8ANeg = Slot7:AMC8:C8:A-
```

```
$S7M1ANeg = Slot7:AMC8:M1:A-
$S7M2ANeg = Slot7:AMC8:M2:A-
$S7M3ANeg = Slot7:AMC8:M3:A-
$S7M4ANeg = Slot7:AMC8:M4:A-

//  Slot 7, Danger+
$S7C1DPos = Slot7:AMC8:C1:D+
$S7C2DPos = Slot7:AMC8:C2:D+
$S7C3DPos = Slot7:AMC8:C3:D+
$S7C4DPos = Slot7:AMC8:C4:D+
$S7C5DPos = Slot7:AMC8:C5:D+
$S7C6DPos = Slot7:AMC8:C6:D+
$S7C7DPos = Slot7:AMC8:C7:D+
$S7C8DPos = Slot7:AMC8:C8:D+
$S7M1DPos = Slot7:AMC8:M1:D+
$S7M2DPos = Slot7:AMC8:M2:D+
$S7M3DPos = Slot7:AMC8:M3:D+
$S7M4DPos = Slot7:AMC8:M4:D+

//  Slot 7, Danger-
$S7C1DNeg = Slot7:AMC8:C1:D-
$S7C2DNeg = Slot7:AMC8:C2:D-
$S7C3DNeg = Slot7:AMC8:C3:D-
$S7C4DNeg = Slot7:AMC8:C4:D-
$S7C5DNeg = Slot7:AMC8:C5:D-
$S7C6DNeg = Slot7:AMC8:C6:D-
$S7C7DNeg = Slot7:AMC8:C7:D-
$S7C8DNeg = Slot7:AMC8:C8:D-
$S7M1DNeg = Slot7:AMC8:M1:D-
$S7M2DNeg = Slot7:AMC8:M2:D-
$S7M3DNeg = Slot7:AMC8:M3:D-
$S7M4DNeg = Slot7:AMC8:M4:D-

//  Slot 7, Global Fail
$S7C1GOKF = Slot7:AMC8:C1:GOKF
$S7C2GOKF = Slot7:AMC8:C2:GOKF
$S7C3GOKF = Slot7:AMC8:C3:GOKF
$S7C4GOKF = Slot7:AMC8:C4:GOKF
$S7C5GOKF = Slot7:AMC8:C5:GOKF
$S7C6GOKF = Slot7:AMC8:C6:GOKF
$S7C7GOKF = Slot7:AMC8:C7:GOKF
$S7C8GOKF = Slot7:AMC8:C8:GOKF
$S7M1GOKF = Slot7:AMC8:M1:GOKF
$S7M2GOKF = Slot7:AMC8:M2:GOKF
```

```
$S7M3GOKF = Slot7:AMC8:M3:GOKF
$S7M4GOKF = Slot7:AMC8:M4:GOKF

//  Slot 7, ADC Error
$S7C1ADERR = Slot7:AMC8:C1:AD_ERR
$S7C2ADERR = Slot7:AMC8:C2:AD_ERR
$S7C3ADERR = Slot7:AMC8:C3:AD_ERR
$S7C4ADERR = Slot7:AMC8:C4:AD_ERR
$S7C5ADERR = Slot7:AMC8:C5:AD_ERR
$S7C6ADERR = Slot7:AMC8:C6:AD_ERR
$S7C7ADERR = Slot7:AMC8:C7:AD_ERR
$S7C8ADERR = Slot7:AMC8:C8:AD_ERR
$S7M1ADERR = Slot7:AMC8:M1:AD_ERR
$S7M2ADERR = Slot7:AMC8:M2:AD_ERR
$S7M3ADERR = Slot7:AMC8:M3:AD_ERR
$S7M4ADERR = Slot7:AMC8:M4:AD_ERR

//  Slot 7, ADC Stdby
$S7C1ADSTBY = Slot7:AMC8:C1:AD_STBY
$S7C2ADSTBY = Slot7:AMC8:C2:AD_STBY
$S7C3ADSTBY = Slot7:AMC8:C3:AD_STBY
$S7C4ADSTBY = Slot7:AMC8:C4:AD_STBY
$S7C5ADSTBY = Slot7:AMC8:C5:AD_STBY
$S7C6ADSTBY = Slot7:AMC8:C6:AD_STBY
$S7C7ADSTBY = Slot7:AMC8:C7:AD_STBY
$S7C8ADSTBY = Slot7:AMC8:C8:AD_STBY
$S7M1ADSTBY = Slot7:AMC8:M1:AD_STBY
$S7M2ADSTBY = Slot7:AMC8:M2:AD_STBY
$S7M3ADSTBY = Slot7:AMC8:M3:AD_STBY
$S7M4ADSTBY = Slot7:AMC8:M4:AD_STBY

//  Slot 7, ADC-PLL_LockErr
$S7C1ADLOCK = Slot7:AMC8:C1:AD_LOCK
$S7C2ADLOCK = Slot7:AMC8:C2:AD_LOCK
$S7C3ADLOCK = Slot7:AMC8:C3:AD_LOCK
$S7C4ADLOCK = Slot7:AMC8:C4:AD_LOCK
$S7C5ADLOCK = Slot7:AMC8:C5:AD_LOCK
$S7C6ADLOCK = Slot7:AMC8:C6:AD_LOCK
$S7C7ADLOCK = Slot7:AMC8:C7:AD_LOCK
$S7C8ADLOCK = Slot7:AMC8:C8:AD_LOCK
$S7M1ADLOCK = Slot7:AMC8:M1:AD_LOCK
$S7M2ADLOCK = Slot7:AMC8:M2:AD_LOCK
$S7M3ADLOCK = Slot7:AMC8:M3:AD_LOCK
$S7M4ADLOCK = Slot7:AMC8:M4:AD_LOCK
```

```
//  Slot 7, ADC_TxErr
$S7C1ADTXERR = Slot7:AMC8:C1:AD_TXERR
$S7C2ADTXERR = Slot7:AMC8:C2:AD_TXERR
$S7C3ADTXERR = Slot7:AMC8:C3:AD_TXERR
$S7C4ADTXERR = Slot7:AMC8:C4:AD_TXERR
$S7C5ADTXERR = Slot7:AMC8:C5:AD_TXERR
$S7C6ADTXERR = Slot7:AMC8:C6:AD_TXERR
$S7C7ADTXERR = Slot7:AMC8:C7:AD_TXERR
$S7C8ADTXERR = Slot7:AMC8:C8:AD_TXERR
$S7M1ADTXERR = Slot7:AMC8:M1:AD_TXERR
$S7M2ADTXERR = Slot7:AMC8:M2:AD_TXERR
$S7M3ADTXERR = Slot7:AMC8:M3:AD_TXERR
$S7M4ADTXERR = Slot7:AMC8:M4:AD_TXERR

//  Slot 7, ADC_DynConfErr
$S7C1ADCFGERR = Slot7:AMC8:C1:AD_CFG_ERR
$S7C2ADCFGERR = Slot7:AMC8:C2:AD_CFG_ERR
$S7C3ADCFGERR = Slot7:AMC8:C3:AD_CFG_ERR
$S7C4ADCFGERR = Slot7:AMC8:C4:AD_CFG_ERR
$S7C5ADCFGERR = Slot7:AMC8:C5:AD_CFG_ERR
$S7C6ADCFGERR = Slot7:AMC8:C6:AD_CFG_ERR
$S7C7ADCFGERR = Slot7:AMC8:C7:AD_CFG_ERR
$S7C8ADCFGERR = Slot7:AMC8:C8:AD_CFG_ERR
$S7M1ADCFGERR = Slot7:AMC8:M1:AD_CFG_ERR
$S7M2ADCFGERR = Slot7:AMC8:M2:AD_CFG_ERR
$S7M3ADCFGERR = Slot7:AMC8:M3:AD_CFG_ERR
$S7M4ADCFGERR = Slot7:AMC8:M4:AD_CFG_ERR

//  Slot 7, Bit_resultFail
$S7C1BITF = Slot7:AMC8:C1:BITF
$S7C2BITF = Slot7:AMC8:C2:BITF
$S7C3BITF = Slot7:AMC8:C3:BITF
$S7C4BITF = Slot7:AMC8:C4:BITF
$S7C5BITF = Slot7:AMC8:C5:BITF
$S7C6BITF = Slot7:AMC8:C6:BITF
$S7C7BITF = Slot7:AMC8:C7:BITF
$S7C8BITF = Slot7:AMC8:C8:BITF
$S7M1BITF = Slot7:AMC8:M1:BITF
$S7M2BITF = Slot7:AMC8:M2:BITF
$S7M3BITF = Slot7:AMC8:M3:BITF
$S7M4BITF = Slot7:AMC8:M4:BITF
```

```
//  Slot 7, NoSample
$S7C1NOSPL = Slot7:AMC8:C1:NOSPL
$S7C2NOSPL = Slot7:AMC8:C2:NOSPL
$S7C3NOSPL = Slot7:AMC8:C3:NOSPL
$S7C4NOSPL = Slot7:AMC8:C4:NOSPL
$S7C5NOSPL = Slot7:AMC8:C5:NOSPL
$S7C6NOSPL = Slot7:AMC8:C6:NOSPL
$S7C7NOSPL = Slot7:AMC8:C7:NOSPL
$S7C8NOSPL = Slot7:AMC8:C8:NOSPL
$S7M1NOSPL = Slot7:AMC8:M1:NOSPL
$S7M2NOSPL = Slot7:AMC8:M2:NOSPL
$S7M3NOSPL = Slot7:AMC8:M3:NOSPL
$S7M4NOSPL = Slot7:AMC8:M4:NOSPL

//  Slot 7, OKLevelFail
$S7C1OKF = Slot7:AMC8:C1:OKF
$S7C2OKF = Slot7:AMC8:C2:OKF
$S7C3OKF = Slot7:AMC8:C3:OKF
$S7C4OKF = Slot7:AMC8:C4:OKF
$S7C5OKF = Slot7:AMC8:C5:OKF
$S7C6OKF = Slot7:AMC8:C6:OKF
$S7C7OKF = Slot7:AMC8:C7:OKF
$S7C8OKF = Slot7:AMC8:C8:OKF
$S7M1OKF = Slot7:AMC8:M1:OKF
$S7M2OKF = Slot7:AMC8:M2:OKF
$S7M3OKF = Slot7:AMC8:M3:OKF
$S7M4OKF = Slot7:AMC8:M4:OKF

//  Slot 7, LinearizationErr
$S7C1LINERR = Slot7:AMC8:C1:LINERR
$S7C2LINERR = Slot7:AMC8:C2:LINERR
$S7C3LINERR = Slot7:AMC8:C3:LINERR
$S7C4LINERR = Slot7:AMC8:C4:LINERR
$S7C5LINERR = Slot7:AMC8:C5:LINERR
$S7C6LINERR = Slot7:AMC8:C6:LINERR
$S7C7LINERR = Slot7:AMC8:C7:LINERR
$S7C8LINERR = Slot7:AMC8:C8:LINERR
$S7M1LINERR = Slot7:AMC8:M1:LINERR
$S7M2LINERR = Slot7:AMC8:M2:LINERR
$S7M3LINERR = Slot7:AMC8:M3:LINERR
$S7M4LINERR = Slot7:AMC8:M4:LINERR
```

```
//  Slot 7, CJT_Err
$S7C1CJERR = Slot7:AMC8:C1:CJERR
$S7C2CJERR = Slot7:AMC8:C2:CJERR
$S7C3CJERR = Slot7:AMC8:C3:CJERR
$S7C4CJERR = Slot7:AMC8:C4:CJERR
$S7C5CJERR = Slot7:AMC8:C5:CJERR
$S7C6CJERR = Slot7:AMC8:C6:CJERR
$S7C7CJERR = Slot7:AMC8:C7:CJERR
$S7C8CJERR = Slot7:AMC8:C8:CJERR
$S7M1CJERR = Slot7:AMC8:M1:CJERR
$S7M2CJERR = Slot7:AMC8:M2:CJERR
$S7M3CJERR = Slot7:AMC8:M3:CJERR
$S7M4CJERR = Slot7:AMC8:M4:CJERR

//  Basic function
$S7BF1 = Slot7:AMC8:Logical:BF1
$S7BF2 = Slot7:AMC8:Logical:BF2
$S7BF3 = Slot7:AMC8:Logical:BF3
$S7BF4 = Slot7:AMC8:Logical:BF4
$S7BF5 = Slot7:AMC8:Logical:BF5
$S7BF6 = Slot7:AMC8:Logical:BF6
$S7BF7 = Slot7:AMC8:Logical:BF7
$S7BF8 = Slot7:AMC8:Logical:BF8
$S7BF9 = Slot7:AMC8:Logical:BF9
$S7BF10 = Slot7:AMC8:Logical:BF10
$S7BF11 = Slot7:AMC8:Logical:BF11
$S7BF12 = Slot7:AMC8:Logical:BF12
$S7BF13 = Slot7:AMC8:Logical:BF13
$S7BF14 = Slot7:AMC8:Logical:BF14
$S7BF15 = Slot7:AMC8:Logical:BF15
$S7BF16 = Slot7:AMC8:Logical:BF16

//  Advanced function
$S7AF1 = Slot7:AMC8:Logical:AF1
$S7AF2 = Slot7:AMC8:Logical:AF2
$S7AF3 = Slot7:AMC8:Logical:AF3
$S7AF4 = Slot7:AMC8:Logical:AF4
$S7AF5 = Slot7:AMC8:Logical:AF5
$S7AF6 = Slot7:AMC8:Logical:AF6
$S7AF7 = Slot7:AMC8:Logical:AF7
$S7AF8 = Slot7:AMC8:Logical:AF8
```

```
// Alarms and Dangers used //
$S7C1ALOWUSED = Slot7:AMC8:Config:C1:A-:Used
$S7C2ALOWUSED = Slot7:AMC8:Config:C2:A-:Used
$S7C3ALOWUSED = Slot7:AMC8:Config:C3:A-:Used
$S7C4ALOWUSED = Slot7:AMC8:Config:C4:A-:Used
$S7C5ALOWUSED = Slot7:AMC8:Config:C5:A-:Used
$S7C6ALOWUSED = Slot7:AMC8:Config:C6:A-:Used
$S7C7ALOWUSED = Slot7:AMC8:Config:C7:A-:Used
$S7C8ALOWUSED = Slot7:AMC8:Config:C8:A-:Used
$S7M1ALOWUSED = Slot7:AMC8:Config:M1:A-:Used
$S7M2ALOWUSED = Slot7:AMC8:Config:M2:A-:Used
$S7M3ALOWUSED = Slot7:AMC8:Config:M3:A-:Used
$S7M4ALOWUSED = Slot7:AMC8:Config:M4:A-:Used
$S7C1AHIGHUSED = Slot7:AMC8:Config:C1:A+:Used
$S7C2AHIGHUSED = Slot7:AMC8:Config:C2:A+:Used
$S7C3AHIGHUSED = Slot7:AMC8:Config:C3:A+:Used
$S7C4AHIGHUSED = Slot7:AMC8:Config:C4:A+:Used
$S7C5AHIGHUSED = Slot7:AMC8:Config:C5:A+:Used
$S7C6AHIGHUSED = Slot7:AMC8:Config:C6:A+:Used
$S7C7AHIGHUSED = Slot7:AMC8:Config:C7:A+:Used
$S7C8AHIGHUSED = Slot7:AMC8:Config:C8:A+:Used
$S7M1AHIGHUSED = Slot7:AMC8:Config:M1:A+:Used
$S7M2AHIGHUSED = Slot7:AMC8:Config:M2:A+:Used
$S7M3AHIGHUSED = Slot7:AMC8:Config:M3:A+:Used
$S7M4AHIGHUSED = Slot7:AMC8:Config:M4:A+:Used
$S7C1DLOWUSED = Slot7:AMC8:Config:C1:D-:Used
$S7C2DLOWUSED = Slot7:AMC8:Config:C2:D-:Used
$S7C3DLOWUSED = Slot7:AMC8:Config:C3:D-:Used
$S7C4DLOWUSED = Slot7:AMC8:Config:C4:D-:Used
$S7C5DLOWUSED = Slot7:AMC8:Config:C5:D-:Used
$S7C6DLOWUSED = Slot7:AMC8:Config:C6:D-:Used
$S7C7DLOWUSED = Slot7:AMC8:Config:C7:D-:Used
$S7C8DLOWUSED = Slot7:AMC8:Config:C8:D-:Used
$S7M1DLOWUSED = Slot7:AMC8:Config:M1:D-:Used
$S7M2DLOWUSED = Slot7:AMC8:Config:M2:D-:Used
$S7M3DLOWUSED = Slot7:AMC8:Config:M3:D-:Used
$S7M4DLOWUSED = Slot7:AMC8:Config:M4:D-:Used
$S7C1DHIGHUSED = Slot7:AMC8:Config:C1:D+:Used
$S7C2DHIGHUSED = Slot7:AMC8:Config:C2:D+:Used
$S7C3DHIGHUSED = Slot7:AMC8:Config:C3:D+:Used
$S7C4DHIGHUSED = Slot7:AMC8:Config:C4:D+:Used
$S7C5DHIGHUSED = Slot7:AMC8:Config:C5:D+:Used
$S7C6DHIGHUSED = Slot7:AMC8:Config:C6:D+:Used
$S7C7DHIGHUSED = Slot7:AMC8:Config:C7:D+:Used
$S7C8DHIGHUSED = Slot7:AMC8:Config:C8:D+:Used
```

```
$S7M1DHIGHUSED = Slot7:AMC8:Config:M1:D+:Used
$S7M2DHIGHUSED = Slot7:AMC8:Config:M2:D+:Used
$S7M3DHIGHUSED = Slot7:AMC8:Config:M3:D+:Used
$S7M4DHIGHUSED = Slot7:AMC8:Config:M4:A+:Used

//  Slot 7, Common Channel Status //
$S7ACNR = Slot7:AMC8:CommonStatus:ACNR
$S7AR   = Slot7:AMC8:CommonStatus:AR
$S7DBP  = Slot7:AMC8:CommonStatus:DBP
$S7SL   = Slot7:AMC8:CommonStatus:SL


00064:3:F = Slot7:AMC8:C1:V
00066:3:F = Slot7:AMC8:C2:V
00068:3:F = Slot7:AMC8:C3:V
00070:3:F = Slot7:AMC8:C4:V
00072:3:F = Slot7:AMC8:C5:V
00074:3:F = Slot7:AMC8:C6:V
00076:3:F = Slot7:AMC8:C7:V
00078:3:F = Slot7:AMC8:C8:V
00080:3:F = Slot7:AMC8:M1:V
00082:3:F = Slot7:AMC8:M2:V
00084:3:F = Slot7:AMC8:M3:V
00086:3:F = Slot7:AMC8:M4:V
00088:3:U = Pack($S7BF1,$S7BF2,$S7BF3,$S7BF4,$S7BF5,$S7BF6,$S7BF7,$S7BF8,
$S7BF9,$S7BF10,$S7BF11,$S7BF12,$S7BF13,$S7BF14,$S7BF15,$S7BF16)
00089:3:U = Pack($S7AF1,$S7AF2,$S7AF3,$S7AF4,$S7AF5,$S7AF6,$S7AF7,$S7AF8,
0,0,0,0,0,0,0,0)
00090:3:U = Pack($S7C1APos,$S7C1ANeg,$S7C1DPos,$S7C1DNeg,$S7C1GOKF,
$S7C1ADERR,$S7C1ADSTBY,$S7C1ADLOCK,$S7C1ADTXERR,$S7C1ADCFGERR,$S7C1BITF,
$S7C1NOSPL,$S7C1OKF,$S7C1LINERR,$S7C1CJERR,0)
00091:3:U = Pack($S7C2APos,$S7C2ANeg,$S7C2DPos,$S7C2DNeg,$S7C2GOKF,
$S7C2ADERR,$S7C2ADSTBY,$S7C2ADLOCK,$S7C2ADTXERR,$S7C2ADCFGERR,$S7C2BITF,
$S7C2NOSPL,$S7C2OKF,$S7C2LINERR,$S7C2CJERR,0)
00092:3:U = Pack($S7C3APos,$S7C3ANeg,$S7C3DPos,$S7C3DNeg,$S7C3GOKF,
$S7C3ADERR,$S7C3ADSTBY,$S7C3ADLOCK,$S7C3ADTXERR,$S7C3ADCFGERR,$S7C3BITF,
$S7C3NOSPL,$S7C3OKF,$S7C3LINERR,$S7C3CJERR,0)
00093:3:U = Pack($S7C4APos,$S7C4ANeg,$S7C4DPos,$S7C4DNeg,$S7C4GOKF,
$S7C4ADERR,$S7C4ADSTBY,$S7C4ADLOCK,$S7C4ADTXERR,$S7C4ADCFGERR,$S7C4BITF,
$S7C4NOSPL,$S7C4OKF,$S7C4LINERR,$S7C4CJERR,0)
00094:3:U = Pack($S7C5APos,$S7C5ANeg,$S7C5DPos,$S7C5DNeg,$S7C5GOKF,
$S7C5ADERR,$S7C5ADSTBY,$S7C5ADLOCK,$S7C5ADTXERR,$S7C5ADCFGERR,$S7C5BITF,
$S7C5NOSPL,$S7C5OKF,$S7C5LINERR,$S7C5CJERR,0)
00095:3:U = Pack($S7C6APos,$S7C6ANeg,$S7C6DPos,$S7C6DNeg,$S7C6GOKF,
$S7C6ADERR,$S7C6ADSTBY,$S7C6ADLOCK,$S7C6ADTXERR,$S7C6ADCFGERR,$S7C6BITF,
$S7C6NOSPL,$S7C6OKF,$S7C6LINERR,$S7C6CJERR,0)
```

```
00096:3:U = Pack($S7C7APos,$S7C7ANeg,$S7C7DPos,$S7C7DNeg,$S7C7GOKF,
$S7C7ADERR,$S7C7ADSTBY,$S7C7ADLOCK,$S7C7ADTXERR,$S7C7ADCFGERR,$S7C7BITF,
$S7C7NOSPL,$S7C7OKF,$S7C7LINERR,$S7C7CJERR,0)

00097:3:U = Pack($S7C8APos,$S7C8ANeg,$S7C8DPos,$S7C8DNeg,$S7C8GOKF,
$S7C8ADERR,$S7C8ADSTBY,$S7C8ADLOCK,$S7C8ADTXERR,$S7C8ADCFGERR,$S7C8BITF,
$S7C8NOSPL,$S7C8OKF,$S7C8LINERR,$S7C8CJERR,0)

00098:3:U = Pack($S7M1APos,$S7M1ANeg,$S7M1DPos,$S7M1DNeg,$S7M1GOKF,
$S7M1ADERR,$S7M1ADSTBY,$S7M1ADLOCK,$S7M1ADTXERR,$S7M1ADCFGERR,$S7M1BITF,
$S7M1NOSPL,$S7M1OKF,$S7M1LINERR,$S7M1CJERR,0)

00099:3:U = Pack($S7M2APos,$S7M2ANeg,$S7M2DPos,$S7M2DNeg,$S7M2GOKF,
$S7M2ADERR,$S7M2ADSTBY,$S7M2ADLOCK,$S7M2ADTXERR,$S7M2ADCFGERR,$S7M2BITF,
$S7M2NOSPL,$S7M2OKF,$S7M2LINERR,$S7M2CJERR,0)

00100:3:U = Pack($S7M3APos,$S7M3ANeg,$S7M3DPos,$S7M3DNeg,$S7M3GOKF,
$S7M3ADERR,$S7M3ADSTBY,$S7M3ADLOCK,$S7M3ADTXERR,$S7M3ADCFGERR,$S7M3BITF,
$S7M3NOSPL,$S7M3OKF,$S7M3LINERR,$S7M3CJERR,0)

00101:3:U = Pack($S7M4APos,$S7M4ANeg,$S7M4DPos,$S7M4DNeg,$S7M4GOKF,
$S7M4ADERR,$S7M4ADSTBY,$S7M4ADLOCK,$S7M4ADTXERR,$S7M4ADCFGERR,$S7M4BITF,
$S7M4NOSPL,$S7M4OKF,$S7M4LINERR,$S7M4CJERR,0)

00102:3:U = Pack($S7C1ALOWUSED,$S7C1AHIGHUSED,$S7C1DLOWUSED,
$S7C1DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00103:3:U = Pack($S7C2ALOWUSED,$S7C2AHIGHUSED,$S7C2DLOWUSED,
$S7C2DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00104:3:U = Pack($S7C3ALOWUSED,$S7C3AHIGHUSED,$S7C3DLOWUSED,
$S7C3DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00105:3:U = Pack($S7C4ALOWUSED,$S7C4AHIGHUSED,$S7C4DLOWUSED,
$S7C4DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00106:3:U = Pack($S7C5ALOWUSED,$S7C5AHIGHUSED,$S7C5DLOWUSED,
$S7C5DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00107:3:U = Pack($S7C6ALOWUSED,$S7C6AHIGHUSED,$S7C6DLOWUSED,
$S7C6DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00108:3:U = Pack($S7C7ALOWUSED,$S7C7AHIGHUSED,$S7C7DLOWUSED,
$S7C7DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00109:3:U = Pack($S7C8ALOWUSED,$S7C8AHIGHUSED,$S7C8DLOWUSED,
$S7C8DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00110:3:U = Pack($S7M1ALOWUSED,$S7M1AHIGHUSED,$S7M1DLOWUSED,
$S7M1DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00111:3:U = Pack($S7M2ALOWUSED,$S7M2AHIGHUSED,$S7M2DLOWUSED,
$S7M2DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00112:3:U = Pack($S7M3ALOWUSED,$S7M3AHIGHUSED,$S7M3DLOWUSED,
$S7M3DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00113:3:U = Pack($S7M4ALOWUSED,$S7M4AHIGHUSED,$S7M4DLOWUSED,
$S7M4DHIGHUSED,0,0,0,0,0,0,0,0,0,0,0,0)

00114:3:U = Pack($S7ACNR,$S7AR,$S7DBP,$S7SL,
0,0,0,0,0,0,0,0,0,0,0,0)


...


PROFINET_CREATE SLOT7 64 114
```

Comme indiqué ci-dessus, dans l'extrait pour une carte AMC8 dans le logement 7 :

- PROFINET est activé dans la section `[GLOBAL]` au moyen de la balise `IS_PROFINET_ACTIVED`.
- Les sorties AMC8 sont définies et organisées dans la section `[MAPPING]`, sur une base logement par logement (carte), au moyen des balises `$S7...`.
- Le mappage du registre Modbus AMC8 est défini dans la section `[MAPPING]`, sur une base logement par logement (carte), au moyen de la syntaxe de mappage de registre Modbus.

  Exemple : `00064:3:F = Slot7:AMC8:C1:V`
  Voir la section 9.2.4 La section MAPPING pour plus d'informations.

**NOTE :** Le mappage du registre Modbus peut utiliser la commande pack (`Pack(...)`) pour combiner des balises `$S7...`.

- Les logements PROFINET sont définis dans la section `[MAPPING]`, à la fin du fichier de configuration, au moyen de balises `PROFINET_CREATE`.

  Exemple : `PROFINET_CREATE SLOT7 64 114`, où
  - `SLOT7` est l'identifiant du logement PROFINET.
    Le numéro d'identification du logement PROFINET peut correspondre au numéro de logement du rack VM600 ou un schéma de numérotation différent peut être utilisé.
  - `64` est la première adresse de registre Modbus mappée sur ce logement PROFINET.
  - `114` est la dernière adresse de registre Modbus mappée sur ce logement PROFINET.
    Les première et dernière adresses de registre Modbus mappées sur un logement PROFINET doivent contenir un bloc de données contigu.

Les structures de données sous-jacentes du logement PROFINET sont définies dans un fichier GSDML (voir la section 10.5 Fichiers GSDML). Le mappage du registre Modbus AMC8 et les définitions des logements PROFINET configurées dans la section `[MAPPING]` du fichier de configuration doivent correspondre aux structures de données définies dans le fichier GSDML.

# ANNEXE E : NOMS DES SYMBOLES

Une liste de noms de symboles (à savoir des balises de description de texte) a été définie pour les cartes MPC4 et AMC8. La liste est disponible sur demande auprès de Meggitt, sous forme d'un fichier texte.

Ces informations peuvent être utilisées avec le logiciel SIMATIC STEP 7, sous forme de symboles, afin de faciliter l'identification et la compréhension des données disponibles à chaque adresse PROFINET.

**NOTE :** Dans le tableau suivant, n (dans la colonne Nom du symbole) définit le numéro de logement du rack VME utilisé dans le système VM600 (entre 3 et 14).

**Tableau E-1 :** Mots réservés (noms de symboles)

| | | Nom du symbole (n est un numéro de logement entre 3 et 14) | Syntaxe des outils de l'extracteur de configuration du fond de panier de MPS | Inv. logique | État normal | Fonction, comme décrite dans le logiciel MPS |
|---|---|---|---|---|---|---|
| MPC | Commun | nMCR | Logement n (MCR) | 0 | 1 | Diagnostics MPC communs - Carte MPC en |
| | | nCMF | Logement n (CMF) | 0 | 0 | Diagnostics MPC communs - Échec de contrôle |
| | | nCOF | Logement n (COF) | 0 | 0 | Canaux communs - Échec de la valeur OK du capteur |
| | | nCPE | Logement n (CPE) | 0 | 0 | Diagnostics MPC communs - Erreur de traitement |
| | | nCIE | Logement n (CIE) | 0 | 0 | Diagnostics du signal commun - Erreur de signal d'entrée |
| | | nCA | Logement n (CA) | 0 | 0 | Alarme commune aux canaux |
| | | nCD | Logement n (CD) | 0 | 0 | Danger commun des canaux |
| | | nCSOL | Logement n (CSOL) | 0 | 0 | Diagnostics Trk/Spd communs - Vitesse hors limite |
| | | nCTL | Logement n (CTL) | 0 | 0 | Diagnostics Trk/Spd communs - Perte du suivi |
| | | nCTOR | Logement n (CTOR) | 0 | 0 | Diagnostics Trk/Spd communs - Suivi hors gamme |
| | | nCDSE | Logement n (CDSE) | 0 | 0 | Diagnostics MPC communs - Erreur de saturation DSP |
| | | nCISE | Logement n (CISE) | 0 | 0 | Diagnostics du signal commun - Erreur de saturation |
| | | nCCME | Logement n (CCME) | 0 | 0 | Diagnostics du signal commun - Dépassement de mode |
| | | nSL | Logement n (SL) | 0 | 0 | Diagnostics MPC communs - État mémorisé/Err. |
| | | nDTM | Logement n (DTM) | 0 | 0 | Commandes communes - Entrée Rehaussement des |
| | | nDBP | Logement n (DBP) | 0 | 0 | Commandes communes - Neutralisation de danger |
| | | nAR | Logement n (AR) | 0 | 0 | Commandes communes - Réinitialisation d'alarme |
| MPC | Canaux de vitesse | nS1A+ | Logement n (S1A+) | 0 | 0 | Canal de vitesse 1 - Alarme + Élevée |
| | | nS1A- | Logement n (nS1A-) | 0 | 0 | Canal de vitesse 1 - Alarme – Faible |
| | | nS1ERR | Logement n (nS1ERR) | 0 | 0 | Canal de vitesse 1 - Invalide |
| | | $OK_SOK_V | Logement n (nS1SOK) | 0 | 1 | Canal de vitesse 1 - Vérification réussie de la valeur OK du |
| | | nS2A+ | Logement n (nS2A+) | 0 | 0 | Canal de vitesse 2 - Alarme + Haute |
| | | nS2A- | Logement n (nS2A-) | 0 | 0 | Canal de vitesse 2 - Alarme – Basse |
| | | nS2ERR | Logement n (nS2ERR) | 0 | 0 | Canal de vitesse 2 - Invalide |
| | | nS2SOK | Logement n (nS2SOK) | 0 | 1 | Canal de vitesse 2 - Vérification réussie de la valeur OK du |

**Tableau E-1 :** Mots réservés (noms de symboles)

| | | Nom du symbole (n est un numéro de logement entre 3 et 14) | Syntaxe des outils de l'extracteur de configuration du fond de panier de MPS | Inv. logique | État normal | Fonction, comme décrite dans le logiciel MPS |
|---|---|---|---|---|---|---|
| MPC | Canal de mesure 1 | nV11A+ | Logement n (V11A+) | 0 | 0 | Canal de mesure 1 - Sortie 1 traitée A+ |
| | | nV11A- | Logement n (V11A-) | 0 | 0 | Canal de mesure 1 - Sortie 1 traitée A- |
| | | nV11D+ | Logement n (V11D+) | 0 | 0 | Canal de mesure 1 - Sortie 1 traitée D+ |
| | | nV11D- | Logement n (V11D-) | 0 | 0 | Canal de mesure 1 - Sortie 1 traitée D- |
| | | nV11ERR | Logement n (V11ERR) | 0 | 0 | Canal de mesure 1 - Sortie 1 traitée Bit d'erreur |
| | | nV12A+ | Logement n (V12A+) | 0 | 0 | Canal de mesure 1 - Sortie 2 traitée A+ |
| | | nV12A- | Logement n (V12A-) | 0 | 0 | Canal de mesure 1 - Sortie 2 traitée A- |
| | | nV12D+ | Logement n (V12D+) | 0 | 0 | Canal de mesure 1 - Sortie 2 traitée D+ |
| | | nV12D- | Logement n (V12D-) | 0 | 0 | Canal de mesure 1 - Sortie 2 traitée D- |
| | | nV12ERR | Logement n (V12ERR) | 0 | 0 | Canal de mesure 1 - Sortie 2 traitée Bit d'erreur |
| | | nV1PGA | Logement n (V1PGA) | 0 | 0 | Canal de mesure 1 - Erreur de saturation PGA |
| | | nV1SOK | Logement n (V1SOK) | 0 | 1 | Canal de mesure 1 - Vérification réussie de la valeur OK du |
| MPC | Canal de mesure 2 | nV21A+ | Logement n (V21A+) | 0 | 0 | Canal de mesure 2 - Sortie 1 traitée A+ |
| | | nV21A- | Logement n (V21A-) | 0 | 0 | Canal de mesure 2 - Sortie 1 traitée A- |
| | | nV21D+ | Logement n (V21D+) | 0 | 0 | Canal de mesure 2 - Sortie 1 traitée D+ |
| | | nV21D- | Logement n (V21D-) | 0 | 0 | Canal de mesure 2 - Sortie 1 traitée D- |
| | | nV21ERR | Logement n (V21ERR) | 0 | 0 | Canal de mesure 2 - Sortie 1 traitée Bit d'erreur |
| | | nV22A+ | Logement n (V22A+) | 0 | 0 | Canal de mesure 2 - Sortie 2 traitée A+ |
| | | nV22A- | Logement n (V22A-) | 0 | 0 | Canal de mesure 2 - Sortie 2 traitée A- |
| | | nV22D+ | Logement n (V22D+) | 0 | 0 | Canal de mesure 2 - Sortie 2 traitée D+ |
| | | nV22D- | Logement n (V22D-) | 0 | 0 | Canal de mesure 2 - Sortie 2 traitée D- |
| | | nV22ERR | Logement n (V22ERR) | 0 | 0 | Canal de mesure 2 - Sortie 2 traitée Bit d'erreur |
| | | nV2PGA | Logement n (V2PGA) | 0 | 0 | Canal de mesure 2 - Erreur de saturation PGA |
| | | nV2SOK | Logement n (V2SOK) | 0 | 1 | Canal de mesure 2 - Vérification réussie de la valeur OK du |
| MPC | Canal de mesure 3 | nV31A+ | Logement n (V31A+) | 0 | 0 | Canal de mesure 3 - Sortie 1 traitée A+ |
| | | nV31A- | Logement n (V31A-) | 0 | 0 | Canal de mesure 3 - Sortie 1 traitée A- |
| | | nV31D+ | Logement n (V31D+) | 0 | 0 | Canal de mesure 3 - Sortie 1 traitée D+ |
| | | nV31D- | Logement n (V31D-) | 0 | 0 | Canal de mesure 3 - Sortie 1 traitée D- |
| | | nV31ERR | Logement n (V31ERR) | 0 | 0 | Canal de mesure 3 - Sortie 1 traitée Bit d'erreur |
| | | nV32A+ | Logement n (V32A+) | 0 | 0 | Canal de mesure 3 - Sortie 2 traitée A+ |
| | | nV32A- | Logement n (V32A-) | 0 | 0 | Canal de mesure 3 - Sortie 2 traitée A- |
| | | nV32D+ | Logement n (V32D+) | 0 | 0 | Canal de mesure 3 - Sortie 2 traitée D+ |
| | | nV32D- | Logement n (V32D-) | 0 | 0 | Canal de mesure 3 - Sortie 2 traitée D- |
| | | nV32ERR | Logement n (V32ERR) | 0 | 0 | Canal de mesure 3 - Sortie 2 traitée Bit d'erreur |
| | | nV3PGA | Logement n (V3PGA) | 0 | 0 | Canal de mesure 3 - Erreur de saturation PGA |
| | | nV3SOK | Logement n (V3SOK) | 0 | 1 | Canal de mesure 3 - Vérification réussie de la valeur OK du |

**Tableau E-1 :** Mots réservés (noms de symboles)

| | | Nom du symbole (n est un numéro de logement entre 3 et 14) | Syntaxe des outils de l'extracteur de configuration du fond de panier de MPS | Inv. logique | État normal | Fonction, comme décrite dans le logiciel MPS |
|---|---|---|---|---|---|---|
| MPC | Canal de mesure 4 | nV41A+ | Logement n (V41A+) | 0 | 0 | Canal de mesure 4 - Sortie 1 traitée A+ |
| | | nV41A- | Logement n (V41A-) | 0 | 0 | Canal de mesure 4 - Sortie 1 traitée A- |
| | | nV41D+ | Logement n (V41D+) | 0 | 0 | Canal de mesure 4 - Sortie 1 traitée D+ |
| | | nV41D- | Logement n (V41D-) | 0 | 0 | Canal de mesure 4 - Sortie 1 traitée D- |
| | | nV41ERR | Logement n (V41ERR) | 0 | 0 | Canal de mesure 4 - Sortie 1 traitée Bit d'erreur |
| | | nV42A+ | Logement n (V42A+) | 0 | 0 | Canal de mesure 4 - Sortie 2 traitée A+ |
| | | nV42A- | Logement n (V42A-) | 0 | 0 | Canal de mesure 4 - Sortie 2 traitée A- |
| | | nV42D+ | Logement n (V42D+) | 0 | 0 | Canal de mesure 4 - Sortie 2 traitée D+ |
| | | nV42D- | Logement n (V42D-) | 0 | 0 | Canal de mesure 4 - Sortie 2 traitée D- |
| | | nV42ERR | Logement n (V42ERR) | 0 | 0 | Canal de mesure 4 - Sortie 2 traitée Bit d'erreur |
| | | nV4PGA | Logement n (V4PGA) | 0 | 0 | Canal de mesure 4 - Erreur de saturation PGA |
| | | nV4SOK | Logement n (V4SOK) | 0 | 1 | Canal de mesure 4 - Vérification réussie de la valeur OK du |
| MPC | Canal de mesure 1 et 2 | nD12A+ | Logement n (D12A+) | 0 | 0 | Canal de mesure 1 et 2 - Sortie 1 traitée A+ |
| | | nD12A- | Logement n (D12A-) | 0 | 0 | Canal de mesure 1 et 2 - Sortie 1 traitée A- |
| | | nD12D+ | Logement n (D12D+) | 0 | 0 | Canal de mesure 1 et 2 - Sortie 1 traitée D+ |
| | | nD12D- | Logement n (D12D-) | 0 | 0 | Canal de mesure 1 et 2 - Sortie 1 traitée D- |
| | | nD12ERR | Logement n (D12ERR) | 0 | 0 | Canal de mesure 1 et 2 - Sortie 2 traitée Bit d'erreur |
| | | nD12PGA | Logement n (D12PGA) | 0 | 0 | Canal de mesure 1 et 2 - Erreur de saturation PGA |
| | | nD12SOK | Logement n (D12SOK) | 0 | 1 | Canal de mesure 1 et 2 - Vérification réussie de la valeur |
| MPC | Canal de mesure 3 et 4 | nD34A+ | Logement n (D34A+) | 0 | 0 | Canal de mesure 3 et 4 - Sortie 1 traitée A+ |
| | | nD34A- | Logement n (D34A-) | 0 | 0 | Canal de mesure 3 et 4 - Sortie 1 traitée A- |
| | | nD34D+ | Logement n (D34D+) | 0 | 0 | Canal de mesure 3 et 4 - Sortie 1 traitée D+ |
| | | nD34D- | Logement n (D34D-) | 0 | 0 | Canal de mesure 3 et 4 - Sortie 1 traitée D- |
| | | nD34ERR | Logement n (D34ERR) | 0 | 0 | Canal de mesure 3 et 4 - Sortie traitée Bit d'erreur |
| | | nD34PGA | Logement n (D34PGA) | 0 | 0 | Canal de mesure 3 et 4 - Erreur de saturation PGA |
| | | nD34SOK | Logement n (D34SOK) | 0 | 1 | Canal de mesure 3 et 4 - Vérification réussie de la valeur |
| MPC | Fonctions de base et avancées | nBF1 | Logement n (BF1) | 0 | 0 | Combinaison de base 1 |
| | | nBF2 | Logement n (BF2) | 0 | 0 | Combinaison de base 2 |
| | | nBF3 | Logement n (BF3) | 0 | 0 | Combinaison de base 3 |
| | | nBF4 | Logement n (BF4) | 0 | 0 | Combinaison de base 4 |
| | | nBF5 | Logement n (BF5) | 0 | 0 | Combinaison de base 5 |
| | | nBF6 | Logement n (BF6) | 0 | 0 | Combinaison de base 6 |
| | | nBF7 | Logement n (BF7) | 0 | 0 | Combinaison de base 7 |
| | | nBF8 | Logement n (BF8) | 0 | 0 | Combinaison de base 8 |
| | | nAF1 | Logement n (AF1) | 0 | 0 | Combinaison avancée 1 |
| | | nAF2 | Logement n (AF2) | 0 | 0 | Combinaison avancée 2 |
| | | nAF3 | Logement n (AF3) | 0 | 0 | Combinaison avancée 3 |
| | | nAF4 | Logement n (AF4) | 0 | 0 | Combinaison avancée 4 |

**Tableau E-1 :** Mots réservés (noms de symboles)

| | | Nom du symbole (n est un numéro de logement entre 3 et 14) | Syntaxe des outils de l'extracteur de configuration du fond de panier de MPS | Inv. logique | État normal | Fonction, comme décrite dans le logiciel MPS |
|---|---|---|---|---|---|---|
| MPC | Fonctions de base et avancées | RLY1A | - | 0 | 0 | RLY1 est sur l'état Alarme (LED RLY1 rouge) |
| | | RLY1ERR | - | 0 | 0 | RLY1 est sur l'état Erreur (LED RLY1 clignotant jaune) |
| | | RLY2A | - | 0 | 0 | RLY2 est sur l'état Alarme (LED RLY2 rouge) |
| | | RLY2ERR | - | 0 | 0 | RLY2 est sur l'état Erreur (LED RLY2 clignotant jaune) |
| | | RLY3A | - | 0 | 0 | RLY3 est sur l'état Alarme (LED RLY3 rouge) |
| | | RLY3ERR | - | 0 | 0 | RLY3 est sur l'état Erreur (LED RLY3 clignotant jaune) |
| | | RLY4A | - | 0 | 0 | RLY4 est sur l'état Alarme (LED RLY4 rouge) |
| | | RLY4ERR | - | 0 | 0 | RLY4 est sur l'état Erreur (LED RLY4 clignotant jaune) |
| | | RLY5A | - | 0 | 0 | RLY5 est sur l'état Alarme (LED RLY5 rouge) |
| | | RLY5ERR | - | 0 | 0 | RLY5 est sur l'état Erreur (LED RLY5 clignotant jaune) |
| | | RLY6A | - | 0 | 0 | RLY6 est sur l'état Alarme (LED RLY6 rouge) |
| | | RLY6ERR | - | 0 | 0 | RLY6 est sur l'état Erreur (LED RLY6 clignotant jaune) |
| | | RLY7A | - | 0 | 0 | RLY7 est sur l'état Alarme (LED RLY7 rouge) |
| | | RLY7ERR | - | 0 | 0 | RLY7 est sur l'état Erreur (LED RLY7 clignotant jaune) |
| | | RLY8A | - | 0 | 0 | RLY8 est sur l'état Alarme (LED RLY8 rouge) |
| | | RLY8ERR | - | 0 | 0 | RLY8 est sur l'état Erreur (LED RLY8 clignotant jaune) |
| | | DSI1 | - | 0 | 0 | Entrée discrète n°1 ('0'=Ouvert, '1'=Court-circuit à la |
| | | DSI2 | - | 0 | 0 | Entrée discrète n°2 ('0'=Ouvert, '1'=Court-circuit à la |
| | | DSI3 | - | 0 | 0 | Entrée discrète n°3 ('0'=Ouvert, '1'=Court-circuit à la |
| | | DSI4 | - | 0 | 0 | Entrée discrète n°4 ('0'=Ouvert, '1'=Court-circuit à la |
| | | AR | - | 0 | 0 | Réinitialisation d'alarme ('0'=Ouvert, '1'=Court-circuit à la |
| | | DB | - | 0 | 0 | Neutralisation de danger ('0'=Ouvert, '1'=Court-circuit à la |
| | | IRCOK | - | 0 | 0 | Commun IRC4 OK |
| AMC | Canal d'entrée 1 | nCH1A+ | Logement n (CH1 A+) | 0 | 0 | Canal d'entrée 1 - Alarme+ |
| | | nCH1A- | Logement n (CH1 A-) | 0 | 0 | Canal d'entrée 1 - Alarme- |
| | | nCH1D+ | Logement n (CH1 D+) | 0 | 0 | Canal d'entrée 1 - Danger+ |
| | | nCH1D- | Logement n (CH1 D-) | 0 | 0 | Canal d'entrée 1 - Danger- |
| | | nCH1GOKF | Logement n (échec global de | 0 | 0 | Canal d'entrée 1 - Échec global de la valeur OK du canal |
| AMC | Canal d'entrée 2 | nCH2A+ | Logement n (CH2 A+) | 0 | 0 | Canal d'entrée 2 - Alarme+ |
| | | nCH2A- | Logement n (CH2 A-) | 0 | 0 | Canal d'entrée 2 - Alarme- |
| | | nCH2D+ | Logement n (CH2 D+) | 0 | 0 | Canal d'entrée 2 - Danger+ |
| | | nCH2D- | Logement n (CH2 D-) | 0 | 0 | Canal d'entrée 2 - Danger- |
| | | nCH2GOKF | Logement n (échec global de | 0 | 0 | Canal d'entrée 2 - Échec global de la valeur OK du canal |
| AMC | Canal d'entrée 3 | nCH3A+ | Logement n (CH3 A+) | 0 | 0 | Canal d'entrée 3 - Alarme+ |
| | | nCH3A- | Logement n (CH3 A-) | 0 | 0 | Canal d'entrée 3 - Alarme- |
| | | nCH3D+ | Logement n (CH3 D+) | 0 | 0 | Canal d'entrée 3 - Danger+ |
| | | nCH3D- | Logement n (CH3 D-) | 0 | 0 | Canal d'entrée 3 - Danger- |
| | | nCH3GOKF | Logement n (échec global de | 0 | 0 | Canal d'entrée 3 - Échec global de la valeur OK du canal |
| AMC | Canal d'entrée 4 | nCH4A+ | Logement n (CH4 A+) | 0 | 0 | Canal d'entrée 4 - Alarme+ |
| | | nCH4A- | Logement n (CH4 A-) | 0 | 0 | Canal d'entrée 4 - Alarme- |
| | | nCH4D+ | Logement n (CH4 D+) | 0 | 0 | Canal d'entrée 4 - Danger+ |
| | | nCH4D- | Logement n (CH4 D-) | 0 | 0 | Canal d'entrée 4 - Danger- |
| | | nCH4GOKF | Logement n (échec global de | 0 | 0 | Canal d'entrée 4 - Échec global du OK du canal |

**Tableau E-1 :** Mots réservés (noms de symboles)

| | | Nom du symbole (n est un numéro de logement entre 3 et 14) | Syntaxe des outils de l'extracteur de configuration du fond de panier de MPS | Inv. logique | État normal | Fonction, comme décrite dans le logiciel MPS |
|---|---|---|---|---|---|---|
| AMC | Canal d'entrée 5 | nCH5A+ | Logement n (CH5 A+) | 0 | 0 | Canal d'entrée 5 - Alarme+ |
| | | nCH5A- | Logement n (CH5 A-) | 0 | 0 | Canal d'entrée 5 - Alarme- |
| | | nCH5D+ | Logement n (CH5 D+) | 0 | 0 | Canal d'entrée 5 - Danger+ |
| | | nCH5D- | Logement n (CH5 D-) | 0 | 0 | Canal d'entrée 5 - Danger- |
| | | nCH5GOKF | Logement n (échec global du | 0 | 0 | Canal d'entrée 5 - Échec global du OK du canal |
| AMC | Canal d'entrée 6 | nCH6A+ | Logement n (CH2 A+) | 0 | 0 | Canal d'entrée 2 - Alarme+ |
| | | nCH6A- | Logement n (CH2 A-) | 0 | 0 | Canal d'entrée 2 - Alarme- |
| | | nCH6D+ | Logement n (CH2 D+) | 0 | 0 | Canal d'entrée 2 - Danger+ |
| | | nCH6D- | Logement n (CH2 D-) | 0 | 0 | Canal d'entrée 2 - Danger- |
| | | nCH6GOKF | Logement n (échec global du | 0 | 0 | Canal d'entrée 2 - Échec global du OK du canal |
| AMC | Canal d'entrée 7 | nCH7A+ | Logement n (CH7 A+) | 0 | 0 | Canal d'entrée 7 - Alarme+ |
| | | nCH7A- | Logement n (CH7 A-) | 0 | 0 | Canal d'entrée 7 - Alarme- |
| | | nCH7D+ | Logement n (CH7 D+) | 0 | 0 | Canal d'entrée 7 - Danger+ |
| | | nCH7D- | Logement n (CH7 D-) | 0 | 0 | Canal d'entrée 7 - Danger- |
| | | nCH7GOKF | Logement n (échec global du | 0 | 0 | Canal d'entrée 7 - Échec global du OK du canal |
| AMC | Canal d'entrée 8 | nCH8A+ | Logement n (CH8 A+) | 0 | 0 | Canal d'entrée 8 - Alarme+ |
| | | nCH8A- | Logement n (CH8 A-) | 0 | 0 | Canal d'entrée 8 - Alarme- |
| | | nCH8D+ | Logement n (CH8 D+) | 0 | 0 | Canal d'entrée 8 - Danger+ |
| | | nCH8D- | Logement n (CH8 D-) | 0 | 0 | Canal d'entrée 8 - Danger- |
| | | nCH8GOKF | Logement n (échec global du | 0 | 0 | Canal d'entrée 8 - Échec global du OK du canal |
| AMC | Canal d'entrée 1 | nAF1 | Logement n (AF1) | 0 / 1 | 0 / 1 | Combinaison logique avancée - Fonction 1 |
| | | nAF2 | Logement n (AF2) | 0 / 1 | 0 / 1 | Combinaison logique avancée - Fonction 2 |
| | | nAF3 | Logement n (AF3) | 0 / 1 | 0 / 1 | Combinaison logique avancée - Fonction 3 |
| | | nAF4 | Logement n (AF4) | 0 / 1 | 0 / 1 | Combinaison logique avancée - Fonction 4 |
| | | nAF5 | Logement n (AF5) | 0 / 1 | 0 / 1 | Combinaison logique avancée - Fonction 5 |
| | | nAF6 | Logement n (AF6) | 0 / 1 | 0 / 1 | Combinaison logique avancée - Fonction 6 |
| | | nAF7 | Logement n (AF7) | 0 / 1 | 0 / 1 | Combinaison logique avancée - Fonction 7 |
| | | nAF8 | Logement n (AF8) | 0 / 1 | 0 / 1 | Combinaison logique avancée - Fonction 8 |
| AMC | Commun | nCSANR | Logement n (état commun AMC hors fonction) | 0 | 0 | État commun - AMC hors fonction |
| | | nCSSL | Logement n (état commun état mémorisé) | 0 | 0 | État commun - État mémorisé |
| | | nCSDB | Logement n (état commun neutralisation de danger) | 0 | 0 | État commun - Neutralisation de danger |
| | | nCSAR | Logement n (état commun réinitialisation d'alarme) | 0 | 0 | État commun - Réinitialisation d'alarme |
| AMC | Canal d'entrée 1 | nMCH1A+ | Logement n (MCH1 A+) | 0 | 0 | Multi-canal 1 - Alarme+ |
| | | nMCH1A- | Logement n (MCH1 A-) | 0 | 0 | Multi-canal 1 - Alarme- |
| | | nMCH1D+ | Logement n (MCH1 D+) | 0 | 0 | Multi-canal 1 - Danger+ |
| | | nMCH1D- | Logement n (MCH1 D-) | 0 | 0 | Multi-canal 1 - Danger- |
| | | nMCH1GOKF | Logement n (échec global du | 0 | 0 | Multi-canal 1 - Échec global du OK du canal |

**Tableau E-1 :** Mots réservés (noms de symboles)

| | | Nom du symbole (n est un numéro de logement entre 3 et 14) | Syntaxe des outils de l'extracteur de configuration du fond de panier de MPS | Inv. logique | État normal | Fonction, comme décrite dans le logiciel MPS |
|---|---|---|---|---|---|---|
| AMC | Canal d'entrée 2 | nMCH2A+ | Logement n (MCH2 A+) | 0 | 0 | Multi-canal 2 - Alarme+ |
| | | nMCH2A- | Logement n (MCH2 A-) | 0 | 0 | Multi-canal 2 - Alarme- |
| | | nMCH2D+ | Logement n (MCH2 D+) | 0 | 0 | Multi-canal 2 - Danger+ |
| | | nMCH2D- | Logement n (MCH2 D-) | 0 | 0 | Multi-canal 2 - Danger- |
| | | nMCH2GOKF | Logement n (échec global du | 0 | 0 | Multi-canal 2 - Échec global du OK du canal |
| AMC | Canal d'entrée 3 | nMCH3A+ | Logement n (MCH3 A+) | 0 | 0 | Multi-canal 3 - Alarme+ |
| | | nMCH3A- | Logement n (MCH3 A-) | 0 | 0 | Multi-canal 3 - Alarme- |
| | | nMCH3D+ | Logement n (MCH3 D+) | 0 | 0 | Multi-canal 3 - Danger+ |
| | | nMCH3D- | Logement n (MCH3 D-) | 0 | 0 | Multi-canal 3 - Danger- |
| | | nMCH3GOKF | Logement n (échec global du | 0 | 0 | Multi-canal 3 - Échec global du OK du canal |
| AMC | Canal d'entrée 4 | nMCH4A+ | Logement n (MCH4 A+) | 0 | 0 | Multi-canal 4 - Alarme+ |
| | | nMCH4A- | Logement n (MCH4 A-) | 0 | 0 | Multi-canal 4 - Alarme- |
| | | nMCH4D+ | Logement n (MCH4 D+) | 0 | 0 | Multi-canal 4 - Danger+ |
| | | nMCH4D- | Logement n (MCH4 D-) | 0 | 0 | Multi-canal 4 - Danger- |
| | | nMCH4GOKF | Logement n (échec global du | 0 | 0 | Multi-canal 4 - Échec global du OK du canal |
| AMC | Combinaison logique de base – Fonctions | nBF1 | Logement n (BF1) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 1 |
| | | nBF2 | Logement n (BF2) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 2 |
| | | nBF3 | Logement n (BF3) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 3 |
| | | nBF4 | Logement n (BF4) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 4 |
| | | nBF5 | Logement n (BF5) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 5 |
| | | nBF6 | Logement n (BF6) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 6 |
| | | nBF7 | Logement n (BF7) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 7 |
| | | nBF8 | Logement n (BF8) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 8 |
| | | nBF9 | Logement n (BF9) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 9 |
| | | nBF10 | Logement n (BF10) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 10 |
| | | nBF11 | Logement n (BF11) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 11 |
| | | nBF12 | Logement n (BF12) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 12 |
| | | nBF13 | Logement n (BF13) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 13 |
| | | nBF14 | Logement n (BF14) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 14 |
| | | nBF15 | Logement n (BF15) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 15 |
| | | nBF16 | Logement n (BF16) | 0 / 1 | 0 / 1 | Combinaison logique de base - Fonction 16 |