

BAUMÜLLER

**INTERFACE  
CAN**

Notice d'instructions

Situation 17 octobre 1996

**F**

5.95076.03

# BAUMÜLLER

## INTERFACE CAN

### Notice d'instructions

Situation 17 octobre 1996

5.95076.03

<p>AVANT LA MISE EN SERVICE LIRE ET OBSERVER LA DESCRIPTION TECHNIQUE ET LES CONSIGNES DE SÉCURITÉ</p>
--

Cette notice d'instructions comprend toutes les informations nécessaires en vue d'une utilisation conforme à l'emploi prévu des produits qui y sont décrits. Elle s'adresse à du personnel technique dûment qualifié disposant d'une formation spéciale et entièrement familiarisé avec toutes les marques d'avertissement et mesures d'entretien. Ces appareils ont été fabriqués selon l'état actuel de la technique et offrent une grande sécurité de fonctionnement. Leur installation et mise en service ne comporte aucun danger et ils marchent sans problème dans la mesure où les consignes de sécurité sont respectées.

La mise en service sera interdite tant que la conformité de la machine où ces constituants seront installés avec les dispositions de la directive machines européenne n'est pas établie.

Avec la remise de la présente description technique et notice d'instructions toutes les descriptions antérieures se rapportant à ce même produit deviennent caduques. Dans le cadre du développement ultérieur de ses produits, la société Baumüller se réserve le droit d'en modifier le maniement et les caractéristiques techniques.

**Adresse du fabricant** Baumüller Nürnberg GmbH  
**et de livraison:** Ostendstr. 80  
90482 Nürnberg  
Téléphone (0911) 5432 - 0                      Télécopie (0911) 5432 - 130

**Copyright:** Toute copie ou reproduction de cette description technique et notice d'instructions est interdite sauf autorisation expresse.

**Pays d'origine:** Fabriqué en République fédérale d'Allemagne

**Date de fabrication:** découle du numéro de fabrication de l'appareil ou du moteur respectif.

## TABLE DES MATIÈRES

<b>1</b>	<b>Caractéristiques techniques .....</b>	<b>1</b>
1.1	Généralités .....	1
1.2	Caractéristiques techniques de la carte optionnelle .....	3
<b>2</b>	<b>Installation .....</b>	<b>5</b>
2.1	Affectation des contacts sur les connecteurs .....	5
2.2	Accessoires .....	8
<b>3</b>	<b>Communication .....</b>	<b>9</b>
3.1	Configuration .....	9
3.1.1	Vitesse de transmission .....	9
3.1.2	Adresse-BUS .....	9
3.2	Transmission des données de processus .....	10
3.2.1	Mot de commande .....	11
3.2.2	Mot d'état .....	11
3.2.3	Valeur de consigne .....	12
3.2.4	Valeurs réelles .....	12
3.2.5	Messages de DIFFUSION (broadcast) .....	13
3.3	Transmission des données utilitaires .....	15
3.3.1	Messages PA .....	15
3.3.2	Message PE .....	16
3.3.3	Paramètres .....	18
3.3.4	Objets .....	20
3.4	Liste d'objets .....	22
3.5	Surveillance de la communication .....	25
3.6	Déverrouillage de la communication .....	26
3.7	Indication sur les transmissions cycliques .....	27



# 1 CARACTÉRISTIQUES TECHNIQUES

## 1.1 Généralités

Une caractéristique essentielle des systèmes numériques d'entraînement est la capacité d'exécuter, en plus de la régulation, un grand nombre de fonctions en partie complexes et de communiquer avec d'autres systèmes intelligents (système de commande, systèmes d'entraînement,...).

En plus du logiciel, ces fonctions exigent des adaptations correspondantes de l'équipement: entrées et sorties optiques, saisies analogiques, analyses de capteurs, ensembles informatiques pour la communication et les fonctions supérieures.

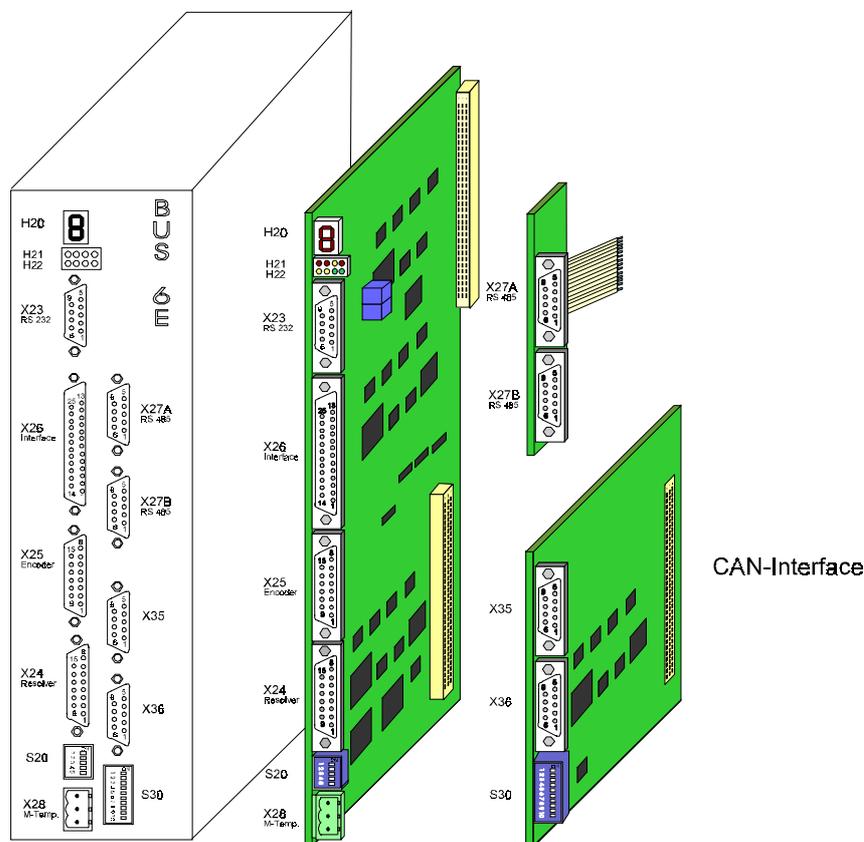
C'est ainsi par ex. qu'un entraînement numérique doit aussi bien pouvoir remplacer un appareil analogique (consigne de vitesse de rotation analogique +/-10 V) qu'être prévu pour de futurs protocoles de communication.

Cela ne peut s'obtenir que grâce à une conception modulaire de l'entraînement:

Le régulateur d'entraînement s'adapte donc, avec des cartes d'extension enfichables, aux différentes exigences.

En plus d'autres platines d'extension, nous offrons une carte optionnelle pour le raccordement à un bus CAN.

Cette carte additionnelle relie le régulateur à des ordinateurs ou à des Automates Programmables Industriels.



Exemple d'utilisation: extension d'un régulateur avec une carte optionnelle Interface CAN

Par le coupleur BUS CAN on peut exploiter l'ensemble des fonctions du régulateur au moyen de la transmission des données de processus et utilitaires. A partir d'un autre noeud CAN (par ex. CAN-01/02/03 pour OMEGA, OMEGA drive-line), c'est possible de réaliser par l'intermédiaire du protocole fixé la présélection cyclique de consignes (transmission des données de processus) et le réglage des paramètres dans le régulateur (transmission des données utilitaires).

Le coupleur BUS CAN dispose de deux fiches (reliées entre elles de manière interne) à séparation galvanique pour le raccordement avec le BUS CAN.

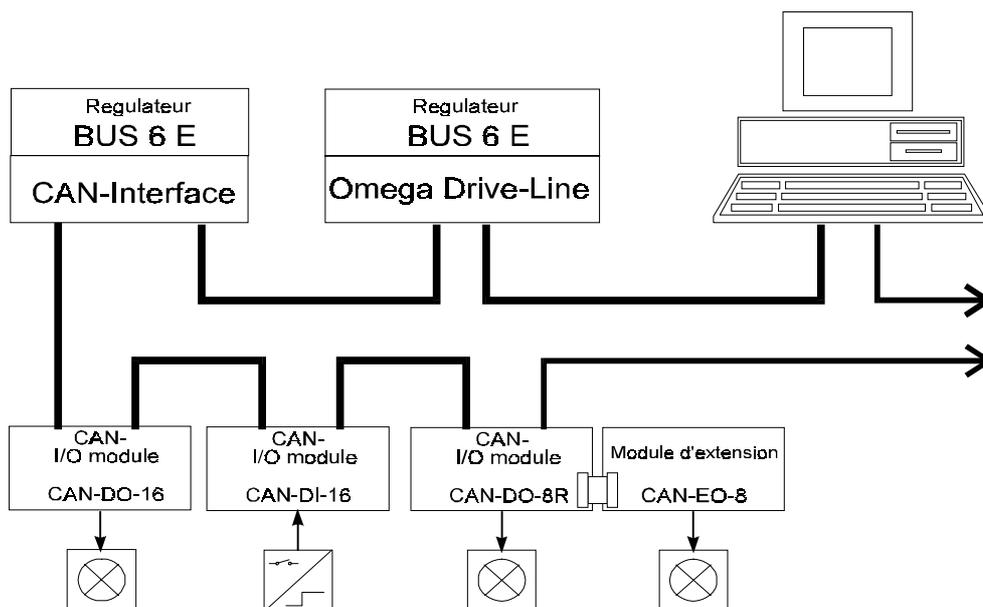
La vitesse de transmission des données est réglable jusqu'à 1 Mbits/s.

### Propriétés

- système de bus sériel asynchrone
- temps réel possible (maximum 1 Mbits/s pour une extension de bus de 40 m),
- diffusion / multicast et communication point à point
- puissance de la détection et du traitement des défauts
- fiabilité élevée (distance Hamming = 6),
- multimaître,
- attribution prioritaire de bus
- temps de latence maximale garantie pour les messages à haute priorité
- standardisation internationale

### Structure de bus

Dans le BUS CAN rattaché, on règle la même vitesse de transmission des données pour tous les participants. Aux deux extrémités le BUS CAN doit être terminé avec une résistance terminale de BUS (124 ohms).



## 1.2 Caractéristiques techniques de la carte optionnelle

En liaison avec la carte optionnelle Interface CAN, le système d'entraînement est un ensemble participant au bus CAN qui dispose d'une bonne communication.

Unité centrale CPU	80C32, 16 MHz
Mémoire	38 kByte RAM, 64 kByte EPROM
Contrôleur CAN	PCA82C200
Couche physique	ISO 11898
Vitesse	max. 1 Mbit/s
Tension de fonctionnement	+5 V (alimenté par la carte de régulation)
Séparation de potentiel	optocoupleur, convertisseur CC /CC
Connecteurs vers le bus CAN	connecteurs mâle et femelle SUB-D à 9 pôles

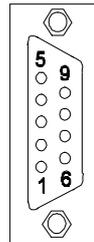


## 2 INSTALLATION

### 2.1 Affectation des contacts sur les connecteurs

- X 35 (la carte CAN sur régulateur BUS 6E)

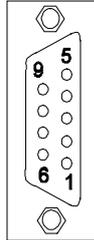
Connecteur mâle sub-D



Pin n°	Attribution
1	pas attribué
2	CAN-LOW
3	TERRE
4	pas attribué
5	pas attribué
6	pas attribué
7	CAN-HIGH
8	pas attribué
9	pas attribué

- X 36 (la carte CAN sur régulateur BUS 6E)

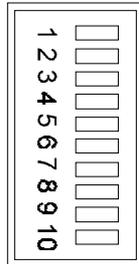
Connecteur femelle sub-D



Pin Nr.	Belegung
1	pas attribué
2	CAN-LOW
3	TERRE
4	pas attribué
5	pas attribué
6	pas attribué
7	CAN-HIGH
8	pas attribué
9	pas attribué

- Commutateur DIP

S20



Nummer	Belegung
1	réglage vitesse transmission bit 0 (HORS 1Mbauds; EN 250 kBauds)
2	réservé, doit être EN
3	réservé
4	réservé
5	réservé
6	réservé
7	bit 0 de l'adresse BUS
8	bit 1 de l'adresse BUS
9	bit 2 de l'adresse BUS
10	bit 3 de l'adresse BUS

**2.2 Accessoires**

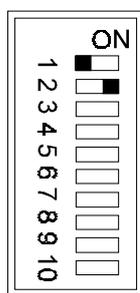
### 3 COMMUNICATION

#### 3.1 Configuration

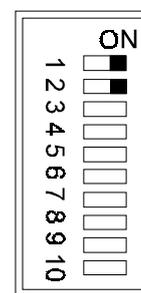
##### 3.1.1 Vitesse de transmission

On peut régler une vitesse de transmission des données de 250 kbits/s ou d'un 1 Mbits/s au moyen du commutateur DIP S20. Le réglage doit se faire le régulateur hors circuit. Un changement de la position du commutateur DIP pendant le fonctionnement est sans effet.

La faible vitesse est nécessaire si d'autres participants au BUS CAN ne supporte pas 1Mbit/s ou si la longueur du BUS ne permet pas 1 Mbit/s.



1 Mbit/s



250 kBit/s

##### 3.1.2 Adresse-BUS

A travers les commutateurs DIP 7 à 10 on règle en binaire l'adresse de BUS (n° d'axe) du coupleur BUS CAN.

Le réglage doit se faire quand le régulateur est hors circuit. Un changement de la position du commutateur DIP pendant le fonctionnement est sans effet.

Le réglage de l'adresse ne dépend pas de l'ordre de succession des noeuds CAN dans le câblage de BUS. Chaque adresse ne doit être attribuée qu'une seule fois dans le BUS CAN rattaché. Une adresse ne peut donc pas être attribuées deux ou trois fois. Pourtant on n'est pas obligé d'attribuer les adresses de bus sans lacune dans l'ordre croissant.



Adresse 0



Adresse 1



Adresse 15

### 3.2 Transmission des données de processus

C'est une transmission cyclique du mot de commande, du mot d'état ainsi que des valeurs de consignes et des valeurs réelles. Les intervalles minimaux du temps de transmission sont les suivants: consigne0 (valeur réelle0): 1 ms; consigne1 (valeur réelle1): 2 ms; consigne2 (valeur réelle2): 4 ms et consigne3 (valeur réelle3): 8 ms. Le mot de commande est pris en charge après 1 ms au maximum.

Le mot de commande correspond au standard DRIVECOM. Il a le numéro de paramètre 120 dans le régulateur. Le mot d'état est délivré par le régulateur; il a le numéro de paramètre 121.

Tous les numéros de paramètres du régulateur peuvent être attribué aux 4 valeurs de consignes et aux 4 valeurs réelles. La liste des numéros de paramètres des consignes et des valeurs réelles doit être définie avec les messages PA ou PE et elle peut être mise en mémoire dans le régulateur.

L'émission cyclique des valeurs réelles peut se définir à travers l'objet 5FF2\_h.. A partir de la version 3.01 les temps de cycle peuvent être sauvegardée dans le régulateur.

Abréviations utilisées dans les listes des messages de transmission des données de processus suivantes :

_h	= valeur en hexadécimal	
_b	= valeur en binaire	
IDB 0	= identificateur octet 0	(v. structure d'un message)
IDB 1	= identificateur bits 2 - 0	( " )
RB	= bit remote	( " )
DLC	= nombre d'octets de données	( " )
DB 0 - DB 7	= octets de données 0 à 7	( " )

Par adresse de bus on entend l'adresse (0 à 15) du coupleur CAN, réglée à l'aide des commutateurs DIP 7 à 10.

### 3.2.1 Mot de commande

**Message "Mot de commande"** (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1
00_h + adresse BUS	001_b	0	2	Mot de commande ID n° 120	

Par le mot de commande, qui a la longueur fixe de 2 octets, on peut commander entre autres le déverrouillage du régulateur, l'arrêt rapide ainsi que tous les modes de contrôle. Dans la Description Technique et la Notice de Service du régulateur on trouvera une description complète des fonctions du mot de commande.

Si le message du mot de commande entraîne un changement d'état dans le régulateur, ce dernier renvoie en réponse le nouvel état dans le message du mot d'état.

### 3.2.2 Mot d'état

**Message "Mot d'état"** (venant du régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1
10_h + adresse BUS	001_b	0	2	Mot d'état ID n° 121	

Le mot d'état, qui a la longueur fixe de 2 octets, indique entre autres l'aptitude à fonctionner et le déverrouillage du régulateur. On trouvera une description complète des fonctions du mot d'état dans la Description du régulateur.

**Demande de Mot d'état** (vers le régulateur)

IDB 0	IDB 1	RB	DLC
10_h + Adresse BUS	001_b	1	0

Par ce message on peut demander l'état actuel du mot d'état du régulateur. Ce dernier répond par le message de Mot d'état (voir ci-dessus).

### 3.2.3 Valeur de consigne

#### Message de Consigne (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3
20_h + Adresse BUS	n° de consigne	0	longueur en octets	Données de consigne minimum 1 octet, maximum 4 octets			

Le numéro de consigne est dans la plage de 0 à 3.

La longueur de consigne dans le message doit concorder avec le format de ce paramètre dans le régulateur (octet => longueur = 1; mot => longueur = 2; double mot => longueur = 4).

#### Message de valeur réelle (venant du régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3
30_h + Adresse BUS	n° valeur réelle	0	longueur en octets	Données valeur réelle minimum 1 octet, maximal 4 octets			

Le numéro de consigne est dans la plage de 0 à 3.

La longueur de valeur réelle dans le message correspond au format de ce paramètre dans le régulateur (octet => longueur = 1; mot => longueur = 2; double mot => longueur = 4).

### 3.2.4 Valeurs réelles

#### Demande de Valeur réelle (vers le régulateur)

IDB 0	IDB 1	RB	DLC
30_h + Adresse BUS	n° valeur réelle	1	0

Par la demande de Valeur réelle, on demande l'état actuel de celle-ci avec le numéro indiqué dans IDB 1.

Le numéro de valeur réelle est dans la plage de 0 à 3.

Le régulateur répond par le message de Valeur réelle (voir ci-dessus).

### 3.2.5 Messages de DIFFUSION (broadcast)

A côté des messages qui doivent posséder la bonne adresse sur le bus, tous les coupleurs BUS CAN reçoivent aussi des messages de diffusion.

#### Message Mot de commande Diffusion (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 0	DB 1
00_h	000_b	0	3	01_h	Mot de commande ID n°120	

Par le mot de commande, qui a la longueur fixe de 2 octets, on peut générer entre autres le déverrouillage de régulateur, l'arrêt rapide. Dans la Description Technique et la Notice de Service du régulateur on trouvera une description complète des fonctions des mots de commande.

Si le message du mot de commande entraîne un changement d'état dans le régulateur, ce dernier renvoie en réponse le nouvel état dans le message du mot d'état.

#### Demande de Mot d'état de Diffusion (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0
00_h	000_b	0	1	02_h

Par ce message on peut demander l'état actuel du mot d'état du régulateur. Ce dernier répond par le message du Mot d'état.

#### Message de Consigne Diffusion (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 1	DB 2	DB 3	DB 4
00_h	000_b	0	longueur en octets + 2	03_h	n° consigne	Données consignes minimum 1 octet, maximum 4 octets			

Le numéro de consigne est dans la plage de 0 à 3.

La longueur de consigne dans le message doit être conforme au format de ce paramètre dans le régulateur.

format de consigne	DLC
octet	3
mot	4
mot double	6

**Demande de Valeur réelle Diffusion** (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4
00_h	000_b	0	1 + nombre de valeurs	04_h	valeur réelle n° 1	valeur réelle n° 2	valeur réelle n° 3	valeur réelle n° 4

Par cette demande on peut faire la demande des valeurs réelles actuelles 1 à 4. Le numéro du paramètre se trouve dans le byte 1 jusqu'au maximum 4. Lors de la communication à 1 Mbits/s seul deux valeurs réelles peuvent être demandées en même temps.

Le numéro de valeur réelle est dans la plage de 0 à 3. Le régulateur répond par le message de valeur réelle (voir ci-dessus).

### 3.3 Transmission des données utilitaires

Cette transmission sert à l'accès direct à un paramètre du régulateur, à un objet existant dans le coupleur BUS CAN. Le n° de paramètre (n° d'objet) est indiqué aussi au moment de l'accès, il n'a donc pas besoin d'être réglé à l'avance. Les accès d'écriture et de lecture reçoivent une réponse par un message du régulateur.

#### 3.3.1 Messages PA

Ces messages servent à définir et à lire les numéros de paramètres des consignes devant être transmises par le message de consignes (3.2.2). Il faut définir combien de consignes sont à transmettre cycliquement et quel n° de paramètre chacune de ces consignes possède dans le régulateur. Une liste de 4 mots (8 octets) sert à cela:

n° mot	Liste des paramètres PA
0	n° du paramètre de consigne 0
1	n° du paramètre de consigne 1
2	n° du paramètre de consigne 2
3	n° du paramètre de consigne 3

Au moment de fixer les numéros de consignes, on veillera à ce que la consigne 0 ne dépasse pas la longueur de 2 octets. S'il faut transmettre moins de 4 valeurs de consignes, pour les consignes non utilisées, on mettra le n° de paramètre 0000\_h.

#### Ecrire des données PA (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4	DB 5	DB 6	DB 7
80_h + Adresse BUS	100_b	0	8	n° consigne 0		n° consigne 1		n° consigne 2		n° consigne 3	
Liste des paramètres PA											

Avec l'ordre "Ecrire données PA", on règle dans le régulateur une nouvelle liste de numéros de paramètres. La transmission des consignes doit ensuite être validée par l'écriture FFFF\_h sur l'objet 6002\_h (voir ci-dessous). Pour les mises hors tension, le réglage peut être sauvegardé par la mise en mémoire de l'ensemble des paramètres (à travers le paramètre ID 190).

On reçoit alors la "Réponse d'écriture PA" (voir ci-dessous).

#### Réponse d'écriture (write) PA (venant du régulateur)

Quand l'accès d'écriture PA a réussi, la réponse suivante est émise:

IDB 0	IDB 1	RB	DLC	DB 0
80_h + Adresse BUS	101_b	0	1	00_h

En cas d'erreur la réponse suivante est émise:

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2
80_h + Adresse BUS	101_b	0	3	80_h	00_h	41_h

A partir de la version 3.01 la réponse est :

IDB 0	IDB 1	RB	DLC	DB 0
80_h + Adresse BUS	101_b	0	1	80_h

**Lecture de données PA** (vers le régulateur)

IDB 0	IDB 1	RB	DLC
80_h + Adresse BUS	100_b	1	0

Par cet ordre on demande la liste actuelle des numéros de paramètres PA qui est dans le régulateur. Le régulateur émet la Réponse Lecture PA.

**Réponse Lecture PA (read)** (venant du régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4	DB 5	DB 6	DB 7
80_h + Adresse BUS	101_b	0	8	n° consigne 0		n° consigne 1		n° consigne 2		n° consigne 3	
Liste des paramètres PA											

La Réponse Lecture PA contient la liste actuelle des numéros de paramètres PA. Elle est émise en réponse à Lire données PA.

### 3.3.2 Message PE

Ces messages servent à régler et à lire les numéros de paramètres des valeurs réelles devant être transmises par le message de valeurs réelles. Il faut définir combien de valeurs réelles sont à transmettre cycliquement et quel n° de paramètre chacune de ces valeurs réelles possède dans le régulateur. Une liste de 4 mots (8 octets) sert à cela:

n° mot	liste des paramètre
0	n° paramètre valeur réelle 0
1	n° paramètre valeur réelle 1
2	n° paramètre valeur réelle 2
3	n° paramètre valeur réelle 3

Au moment de fixer les numéros de valeurs réelles, on notera que la valeur réelle 0 ne doit pas dépasser la longueur de 2 octets. S'il ne faut pas transmettre 4 valeurs réelles, pour celles non utilisées, on mettra le n° de paramètre 0000\_h.

**Ecrire des données PE** (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4	DB 5	DB 6	DB 7
80_h + Adresse BUS	110_b	0	8	n° de valeur réelle 0		n° de valeur réelle 1		n° de valeur réelle 2		n° de valeur réelle 3	
liste des n° de paramètres PE											

Avec l'ordre "Ecrire données PE", on définit dans le régulateur une nouvelle liste de numéros de paramètres PE. Pour les mises hors tension, le réglage peut être sauvegardé par la mise en mémoire de l'ensemble des paramètres (à travers le paramètre 190).

On reçoit alors la "Réponse Ecriture PE".

**Réponse Ecriture PE (write)** (venant du régulateur)

Quand l'accès d'écriture PE a réussi, la réponse suivante est émise:

IDB 0	IDB 1	RB	DLC	DB 0
80_h Adresse BUS	111_b	0	1	00

En cas d'erreur la réponse suivante est émise:

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2
80_h Adresse BUS	111_b	0	3	80_h	00_h	41_h

A partir de la version 3.01 la réponse est :

IDB 0	IDB 1	RB	DLC	DB 0
80_h Adresse BUS	111_b	0	1	80_h

**Lecture de données PE** (vers le régulateur)

IDB 0	IDB 1	RB	DLC
80_h Adresse BUS	110_b	1	0

Par cet ordre on demande la liste actuelle des numéros de paramètres PE qui est dans le régulateur.

Le régulateur émet la Réponse Lecture PE.

**Réponse Lecture (read) PE** (venant du régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4	DB 5	DB 6	DB 7
80_h + Adresse BUS	111_b	0	8	n° de valeur réelle 0		n° de valeur réelle 1		n° de valeur réelle 2		n° de valeur réelle 3	
liste des n° de paramètres PE											

La Réponse Lecture PE contient la liste actuelle des numéros de paramètres PE. Elle est émise en réponse à Lire données PE.

**3.3.3 Paramètres**

**Ecrire des paramètres** (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4	DB 5	DB 6	DB 7
80_h Adresse BUS	010_b	0	5,6,8	Ordre	élément	n° de paramètre		Données min. 1 octet      max. 4 octets			

Par cet ordre on écrit sur un paramètre dans le régulateur.

Dans "Ordre" se trouve l'ordre d'écriture combiné avec le format de paramètre.

écriture d'un	ordre	DLC	nombre d'octets de données
paramètre octet	08_h	5	5
paramètre mot	09_h	6	6
paramètre double mot	0A_h	8	8

"Elément" contient l'élément de paramètre qui doit être écrit. Pour l'écriture de la valeur du paramètre, l'élément contient 07\_h. Dans "n° de paramètre" on inscrit le n° de paramètre du régulateur.

On reçoit alors la Réponse Ecriture paramètre.

**Réponse Ecriture paramètre** (venant du régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2
80_h Adresse BUS	011_b	0	1 3 en cas d'erreur	résultat	n° d'erreur (en cas d'erreur)	

Ce message est la réponse du régulateur à un ordre d'écriture de paramètre.

Dans le résultat on a 08\_h si l'accès d'écriture de paramètre a réussi. En cas de défaut, on a dans le résultat la valeur 28\_h; et dans DB 1 et DB 2 le numéro de défaut du régulateur (voir Description du régulateur).

**Lecture de paramètre (vers le régulateur)**

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB3
80_h Adresse BUS	010_b	0	4	Format	élément	n° de paramètre	

Par cet ordre on lit un paramètre dans le régulateur. Dans cet ordre on peut indiquer le format du paramètre dans "format".

lecture d'un:	format
paramètre octet	00_h
paramètre mot	01_h
paramètre double mot	02_h

"Elément" contient l'élément de paramètre qui doit être lu. Pour la lecture de la valeur du paramètre, l'élément contient 07\_h. Dans "n° de paramètre" on inscrit le n° de paramètre du régulateur

Le régulateur émet alors la Réponse Lecture paramètre.

**Réponse Lecture paramètre (read) (venant du régulateur)**

La réponse à Lecture paramètre a la forme suivante quand la lecture a réussi:

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4
80_h Adresse BUS	011_b	0	2,3,5	Format	Données min. 1 Byte                      max. 4 Bytes			

Dans "Format" on trouve le format du paramètre lu.

lecture d'un:	Format	DLC
paramètre octet	00_h	2
paramètre mot	01_h	3
paramètre double mot	02_h	5

Le nombre d'octets de données émis correspond au format (voir aussi DLC).

Si l'accès de lecture n'a pas pu s'exécuter, la Réponse à Lecture paramètre a l'allure suivante:

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2
80_h Adresse BUS	011_b	0	3	20_h	n° d'erreur	

Le régulateur délivre un n° d'erreur.

### 3.3.4 Objets

**Ecrire un objet** (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4	DB 5	DB 6	DB 7
80_h Adresse BUS	000_b	0	5,6,8	ordre	sous index	n° objet		Données min. 1 octet      max. 4 octets			

Par cet ordre on écrit un objet du coupleur BUS CAN

Dans "Ordre" l'ordre de lecture est combiné avec le format de l'objet.

écriture d'un:	ordre	DLC	nombre d'octets de données
objet octet	08_h	5	5
objet mot	09_h	6	6
objet double mot	0A_h	8	8

"Sous-index" (subindex) contient le sous-index de l'objet qui doit être écrit. Pour l'écriture de tout l'objet, le sous-objet contient 00\_h. Dans "n° d'objet" on inscrit le n° d'objet BUS CAN (voir ci-dessous Liste d'objets).

On reçoit alors la Réponse Ecriture objet.

**Réponse Ecriture objet** (venant du régulateur)

IDB 0	IDB 1	RB	DLC	DB 0
80_h + Adresse BUS	001_b	0	1	résultat

Ce message est la réponse du régulateur à un ordre d'écriture d'objet.

Dans le résultat on a 00\_h si l'accès d'écriture d'objet a réussi. En cas de défaut, on a dans le résultat la valeur 50\_h.

**Lecture Objet** (vers le régulateur)

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB3
80_h Adresse BUS	000_b	0	4	Format	sous index	n° objet	

Par cet ordre on lit un objet dans le régulateur. Dans cet ordre on doit indiquer le format de l'objet dans "format".

lecture d'un:	Format
objet octet	00_h
objet mot	01_h
objet double mot	02_h

"Sous-index" contient le sous-index de l'objet qui doit être lu. Pour la lecture de tout l'objet, le sous-index contient 00\_h. Dans "n° de l'objet" on inscrit le n° de l'objet du coupleur BUS CAN. (voir liste d'objets ci-dessous).

Le régulateur émet alors la Réponse Lecture objet.

**Réponse Lecture objet (read)** (venant du régulateur)

La réponse à Lecture objet a la forme suivante quand la lecture a réussi:

IDB 0	IDB 1	RB	DLC	DB 0	DB 1	DB 2	DB 3	DB 4
80_h Adresse BUS	001_b	0	2,3,5	00_h	Données min. 1 octet                      max. 4 octets			

Le nombre d'octets envoyés correspond au nombre contenu dans Format de l'objet.

Si l'accès de lecture n'a pas pu s'exécuter, la Réponse à Lecture objet a l'allure suivante:

IDB 0	IDB 1	RB	DLC	DB 0
80_h Adresse BUS	001_b	0	1	50_h

### 3.4 Liste d'objets

Le tableau suivant indique les numéros d'objets admis.

#### INSTRUCTION

On ne doit pas accéder à d'autres numéros d'objets.

n° d'objet	type de donnée	type d'objet	signification
5FF2_h	mot	tableau (array)	temps de cycle de valeur réelle
5FF4_h	mot	Variable	Surveillance de la communication CAN
6002_h	16bits	Variable	Déverrouillage de consigne

L'objet 5FF4\_h n'existe que depuis la version 3.01.

Le contenu des objets ne peut pas être sauvegardé par la mise en mémoire des paramètres qui sont dans le régulateur. Après chaque mise en circuit du régulateur, il faut donc récrire les réglages.

Après un redémarrage du régulateur toutes les consignes du message de consignes PA sont automatiquement déverrouillées. Ceci est la valeur initiale pour l'objet 6002\_h.

A partir de la version 3.01 les temps de cycles de valeur réelle peuvent être sauvegardés dans le régulateur. Ils sont automatiquement chargés lors du démarrage du régulateur

#### Temps de cycle de valeur réelle (5FF2\_h)

Dans l'objet Temps de cycle de valeur réelle on peut régler pour chacune des quatre valeurs réelle (voir transmission des données de processus) avec quel temps de cycle elle doit être émise automatiquement par le régulateur. Sur cet objet, on ne peut pas accéder à l'ensemble de l'objet par l'intermédiaire de Sous-index=0. L'accès doit se faire par Sous-index 1 à 4. Il faut alors régler, pour valeur réelle0, sous-index=1; pour valeur réelle1, sous-index=2, etc

La base de temps est dépendante de la vitesse de transmission du BUS CAN.

- Pour des vitesses jusqu'à 250 kbits/s, elle est de 4 ms
- Pour des vitesses à partir de 500 kbits/s, elle est de 0.5 ms

A partir de la version 3.01, la base est fixe à 0.5 ms.

Si le temps de cycle est mis à 0, la valeur réelle correspondante n'est pas transmise cycliquement. Elle ne peut alors se lire que par l'intermédiaire de la demande de valeur réelle.

attribut d'objet	valeur
numéro d'objet	5FF2_h
type d'objet	tableau (array)
type de donnée	mot (16 bits)
nombre d'éléments	4

sous-index	temps de cycle valeur réelle de
1	valeur réelle 0
2	valeur réelle 1
3	valeur réelle 2
4	valeur réelle 3

Le réglage par défaut ne transmet aucune valeur réelle cycliquement.

#### Temps de surveillance communication CAN(5FF4\_h) (à partir de la version 3.01)

Avec cet objet il est possible de définir le temps de surveillance, après lequel la communication est coupée si aucun message n'est reçu. (voir Surveillance communication)

attribut d'objet	valeur
numéro d'objet	5FF4_h
type d'objet	variable
type de donnée	mot (16 bits)
nombre d'élément	1
signification	Temps de surveillance du BUS CAN en ms

**Déverrouillage de consigne (6002\_h)**

Par cet objet on peut réaliser la validation de consigne pour le maximum des quatre consignes des données de processus. C'est nécessaire surtout après un changement des données PA (voir transmission des données de processus).

L'objet est une variable de 16 bits, où chaque bit permet de valider un octet de la transmission des données de processus. Les deux premiers bits (bits 0 et 1) concernent la validation du mot de commande. Les deux suivants (2 et 3) permettent de valider la première consigne (grandeur de mot). Les bits de poids plus élevé valident les octets des autres consignes (s'il y en a de fixées dans la description des données PA). On obtient la validation complète en écrivant sur l'objet Déverrouillage de Consigne la valeur FFFF\_h.

attribut d'objet	valeur
numéro d'objet	6002_h
type d'objet	variable
type de donnée	mot (16 bits)
nombre d'élément	1

bit	Déverrouillage de
0	mot de commande octet 0
1	mot de commande octet 1
2	consigne 0 octet 0
3	consigne 0 octet 1
4	consigne 1 octet 0
5	consigne 1 octet 1
6	consigne 1 octet 2
7	consigne 1 octet 3
8	consigne 2 octet 0
9	consigne 2 octet 1
10	consigne 2 octet 2
11	consigne 2 octet 3
12	consigne 3 octet 0
13	consigne 3 octet 1
14	consigne 3 octet 2
15	consigne 3 octet 3

Le réglage standard est le suivant: toutes les consignes inscrites dans les données PA sont validées.

### **3.5 Surveillance de la communication**

Dans le régulateur il est possible de surveiller la communication avec la carte optionnelle CAN. Ceci est réalisé avec les paramètres ID 127 à 129 dans le régulateur (Voir description régulateur). Cette surveillance devient active si pendant le temps indiqué au paramètre ID128 aucune communication entre le régulateur et la carte n'a eu lieu. Ceci permet de contrôler le bon fonctionnement de la carte optionnelle. Il est aussi nécessaire de surveiller le bon fonctionnement du BUS CAN en lui-même.

Jusqu'à la version 2 un cycle de communication n'avait lieu que si un message était reçu sur le BUS CAN. Il était alors possible de surveiller le BUS CAN depuis le régulateur. Il n'était par contre pas possible de recevoir des valeurs réelles plus rapidement que les demandes de valeurs réelles ne le permettait

A partir de la version 3.01 la carte optionnelle CAN est équipée d'une surveillance. (objet 5FF4\_h).

Après chaque message CAN reçu la communication est maintenue pendant ce temps et des nouvelles valeurs réelles sont émises du régulateur. Si aucun message n'est reçu sur le BUS pendant cette période, la communication avec le régulateur est interrompue. C'est à cet instant que la surveillance dans le régulateur démarre.

Lors de l'enclenchement c'est la valeur du paramètre ID 128 qui est écrit comme valeur par défaut dans l'objet 5FF4\_h. Un changement par une écriture paramètre sur le régulateur modifie aussi la valeur dans l'objet. En plus cette valeur peut être modifiée par une écriture objet mais cette valeur est perdue lors du déclenchement.

On obtient une surveillance particulièrement puissante en faisant les réglages suivants : ID128 dans le régulateur = 1 ms et l'objet 5FF4 sur la durée maximale entre deux messages vers le régulateur.

### 3.6 Déverrouillage de la communication

A partir de la version 3.01 il est nécessaire de déverrouiller la communication par le BUS CAN. Avant ce déverrouillage la carte optionnelle ignore tous les messages sur le BUS CAN et n'en envoie pas (même si des valeurs réelles doivent être transmises cycliquement). Ce déverrouillage peut se faire de la manière suivante :

1. Envoyer depuis le maître le message de diffusion (identificateur = 000\_h) de longueur 1 octet et de valeur = FF\_h.  
(Ce message est envoyé directement après le démarrage à partir de la version 3.01 de l'Omega-Master)
2. Envoyer le message avec l'identificateur 0AA\_h, de longueur 2 octets = AA\_h; 04\_h. Le déverrouillage sera effectif 200 ms après la réception du message. (Ce message est envoyé par l'Omega Master lors de l'exploitation cyclique du BUS CAN. Ceci permet le déverrouillage d'un régulateur qui démarre avec un retard.)

### 3.7 Indication sur les transmissions cycliques

La transmission de données de processus est conçue comme une communication cyclique. Sur la carte optionnelle, il est possible de définir les temps de cycle pour les valeurs réelles. (objet 5FF2\_h).

Les tableaux suivants permettent de se faire une idée des temps de cycle. Les valeurs sont les temps de cycle les plus court réalisables avec les réglages de vitesse et le nombre de valeurs réelles à transmettre. Il est important que la charge du BUS CAN n'atteigne pas 100% parce que les messages de niveau de priorité plus faible ne sont transmis qu'avec un retard important ou plus transmis du tout.

Temps de cycle minimaux possible si seul des valeurs réelles sont transmises cycliquement :

Baudrate	une valeur réelle				deux valeurs réelles				trois valeurs réelles				quatre valeurs réelles	
	2 octets		4 octets		2 octets		4 octets		2 octets		4 octets		2 octets	
en	ZZ	BL	ZZ	BL	ZZ	BL	ZZ	BL	ZZ	BL	ZZ	BL	ZZ	BL
1000	0.5	14%	0.5	17%	0.5	29%	1.0	18%	1.0	23%	1.0	26%	1.0	36%
500	0.5	25%	0.5	33%	1.0	26%	1.0	35%	1.0	45%	1.0	51%	1.5	40%
250	0.5	54%	0.5	66%	1.0	53%	1.0	67%	1.5	52%	1.5	87%	2.0	52%
125	1.0	53%	1.0	65%	1.5	70%	2.0	68%	2.0	80%	2.5	82%	3.0	70%

abréviation : ZZ: temps de cycle en ms  
BL: env. charge de BUS en %

Temps de cycle minimaux possible si seul des valeurs de consigne et des valeurs réelles sont transmises cycliquement

Baudrate	une consigne / une valeur réelle				deux consignes / deux valeurs réelles				trois consignes / trois valeurs réelles				quatre consignes / quatre valeurs réelles	
	2 octets		4 octets		2 octets		4 octets		2 octets		4 octets		2 octets	
en	ZZ	BL	ZZ	BL	ZZ	BL	ZZ	BL	ZZ	BL	ZZ	BL	ZZ	BL
1000	1.0	13%	1.0	17%	1.0	28%	1.0	18%	1.0	40%	1.0	58%	1.0	53%
500	1.0	27%	1.0	34%	1.0	54%	1.0	69%	1.0	84%	2.0	52%	2.0	58%
250	1.0	53%	1.0	66%	2.0	53%	2.0	69%	2.0	81%	3.0	68%	3.0	72%
125	2.0	53%	2.0	65%	3.0	70%	3.0	86%	4.0	80%	5.0	80%	5.0	84%

abréviation : ZZ: temps de cycle en ms  
BL: env. charge de BUS en %

Si les données DRIVECOM sont utilisées 6044\_h (Vitesse min<sup>-1</sup>) ou 6054\_h (Vitesse en %) les temps de cycle se rallongent de 0.5 ms. En utilisant les données DRIVECOM pour les consignes 6042\_h (Vitesse absolue) ou 6052\_h (Vitesse en %) les temps de cycle se rallonge de 1 ms. Il est donc conseillé de ne pas utiliser ces données mais les valeurs normées internes du régulateur (100% = 10000).