

Notice

Sendix 7058 / 7078 / 7158 / 7178
Codeurs absolus monotour

Sendix 7068 / 7088 / 7168 / 7188
Codeurs absolus multitours



CANopen®

Editeur	Kübler Group, Fritz Kübler GmbH Schubertstr. 47 78054 Villingen-Schwenningen Allemagne www.kuebler.com
Assistance applications	Tél. +49 7720 3903-952 Télécopie +49 7720 21564 support@kuebler.com
N° de document	R60712
Titre du document	Notice, Sendix 7058 / 7078 / 7158 / 7178 - CANopen Sendix 7068 / 7088 / 7168 / 7188 - CANopen
Langue	Français (FR) – La version originale est en langue anglaise
Version de document	28.06.2018, N° de doc. R60712.0003 - Index 3
Copyright	©2018, Kübler Group, Fritz Kübler GmbH
Mentions légales	L'ensemble du contenu de la présente description d'appareil est soumis aux droits d'utilisation et d'auteur de Fritz Kübler GmbH. Toute duplication, modification, utilisation ultérieure ou publication sur d'autres média électroniques ou imprimés, ainsi que leur publication dans l'Internet, n'est permise qu'avec l'autorisation écrite préalable de Fritz Kübler GmbH.

Sommaire

Sommaire

Liste des abréviations

1. Détails techniques et caractéristiques.....	6
1.1 Codeurs multitours/monotour CANopen Série 58X8.....	6
1.2 Le profil de communication CANopen DS 301 V4.2.0.....	6
1.3 Profil d'appareil pour codeur DS 406 V3.1.....	7
1.4 Objectifs de LSS.....	7
1.5 Transmission de données.....	7
1.6 Objets et codes de fonctions dans le Predefined Connection Set.....	8
1.7 Transmission des données process.....	9
2. Installation électrique et bus CAN.....	11
2.1 Tension d'alimentation (alimentation électrique).....	11
2.2 Lignes CANbus.....	11
2.3 Terminaison du bus.....	12
3. Mise à la terre et liaison équipotentielle.....	13
4. Guide de démarrage rapide - Réglages par défaut à la livraison.....	14
4.1 Codeurs avec départ de câble.....	14
4.2 Paramètres de communication.....	14
4.3 Profil Codeur.....	15
5. Objets Emergency.....	17
6. Protocole Heartbeat.....	18
7. Objets du Profil Codeur DS 306.....	19
7.1 Objet 6000h Paramètres de fonctionnement.....	19
7.2 Objet 6001h : nombre de pas de mesure par tour (Résolution).....	20
7.3 Objet 6002h : Nombre total de pas de mesure.....	20
7.4 Objet 6003h : Valeur de prépositionnement.....	20
7.5 Objet 6004h : Valeur de position.....	21
7.6 Objet 6030h : Valeur de vitesse.....	21
7.7 Objet 6040h : Valeur d'accélération.....	21
7.8 Objet 6200h : Temporisateur cyclique.....	22
7.9 Objet 6500h : Affichage de l'état de fonctionnement.....	22
7.10 Objet 6502h : Nombre de tours multitours.....	22
7.11 Objet 6503h : Alarmes.....	23
7.12 Objet 6504h : Alarmes supportées.....	23

7.13	Objet 6505h : Avertissements.....	23
7.14	Objet 6506h : Avertissements supportés	24
7.15	Objet 6400h : Registre d'état de la zone de travail 2 valeurs	24
7.16	Objet 2100h : Vitesse de transmission.....	25
7.17	Objet 2101h : Adresse de nœud.....	25
7.18	Objet 2102h : Terminaison de bus CAN OFF/ON	26
7.19	Objet 2103h : Version du firmware flash	26
7.20	Objet 2105h : Save All Bus Parameters (enregistrement de tous les paramètres du bus)	
	27	
7.21	Objet 2110h : Données de configuration du capteur	27
7.22	Objet 2120,4h : Température courante capteur de position *	27
7.23	Objet 2120,2h : Température courante limite inférieure-Capteur de position	28
7.24	Objet 2120,3h : Température courante limite supérieure-Capteur de position	28
7.25	Objet 2130h : Pas de mesure du codeur.....	29
7.26	Objet 2140h : Mémoire utilisateur (16 octets)	29
7.27	Objet 2150h : Historique de la température	29
8.	Gestion du réseau	31
9.	Instructions NMT	32
10.	Glossaire	33

Liste des abréviations

Abréviation	Signification
CAL	CAN Application Layer. Couche d'application (couche 7) dans le modèle de communication CAN
CAN	Controller Area Network
CiA	CAN in Automation. Association internationale des utilisateurs et constructeurs de produits CAN
CMS	CAN Message Specification. Elément de service de CAL
COB	Communication Object. Unité de transport dans le réseau CAN (message CAN). Les données sont envoyées sur le réseau dans un COB.
COB-ID	Identifiant COB Désignation univoque d'un message CAN. L'identifiant détermine la priorité des COB sur le réseau.
DBT	Distributor. Elément de service de CAL, responsable de l'allocation dynamique des identifiants.
DS	Draft Standard ; Projet de norme
DSP	Draft Standard Proposal ; Proposition de projet de norme
ID	Identifiant, voir ID COB
LMT	Layer Management. Elément de service de CAL, responsable de la configuration des paramètres dans les différentes couches du modèle de communication.
LSB	Least Significant Bit/Byte ; Bit/octet de poids le plus faible.
MSB	Most Significant Bit/Byte ; Bit/octet de poids le plus fort NMT Network Management. Elément de service de CAL, responsable de l'initialisation, de la configuration et de la gestion des erreurs dans le réseau.
MT	Codeur multiteurs
OSI	Open Systems Interconnection. Modèle de couches pour la description des zones fonctionnelles dans un système de communication de données.
PDO	Process Data Object. Objet pour l'échange de données process.
RTR	Remote Transmission Request ; Télégramme de requête de données
SDO	Service Data Object ; Objet de communication permettant au maître d'accéder au répertoire d'objets d'un nœud.
SYNC	Télégramme de synchronisation. Les participants au bus répondent à l'instruction SYNC en retournant la valeur de leur process.

1. Détails techniques et caractéristiques

1.1 Codeurs multitours/monotour CANopen Série 58X8

Les codeurs CANopen de la série 7068/88 supportent le profil de communication CANopen le plus récent selon **DS 301 V4.02**. Ils implémentent en outre des profils d'appareil spécifiques tels que le profil codeur **DS 406 V3.1**. Les modes opératoires suivants sont disponibles : Polled Mode, Cyclic Mode, Sync Mode et un protocole Sync haute résolution.

Le bus CAN permet en outre la programmation de facteurs d'échelle, de valeurs de prépositionnement, de valeurs de fin de course et de nombreux autres paramètres. A la mise sous tension, tous les paramètres, mémorisés au préalable dans une mémoire non-volatile pour les protéger contre toute coupure de courant, sont chargés depuis une EEPROM. Les valeurs de sortie suivantes peuvent se combiner librement sous la forme de **PDO** (mappage des PDO) : **position**, **vitesse**, **accélération**, ainsi que l'état des quatre **fins de course**.

Ces codeurs sont disponibles avec **sortie par câble** ; l'adresse d'appareil et la vitesse de transmission se règlent par logiciel. Ces modèles sont équipés d'un coupleur en T intégré permettant une installation aisée.

1.2 Le profil de communication CANopen DS 301 V4.2.0

CANopen constitue une interface utilisateur homogène et simplifie ainsi la réalisation de systèmes réunissant les appareils les plus variés. CANopen est optimisé pour l'échange rapide de données dans des systèmes en temps réel et dispose de différents profils d'appareil qui ont été standardisés. L'association de fabricants et d'utilisateurs CAN in Automation (CiA) est responsable de l'élaboration et de la normalisation des profils correspondants.

CANopen offre

- un accès confortable à tous les paramètres des appareils
- une auto-configuration du réseau et des appareils
- la synchronisation des appareils dans le réseau
- un trafic des données process cyclique et lié à des événements
- une lecture et une émission simultanées de données

CANopen fait appel à quatre objets de communication (COB) ayant des caractéristiques différentes

- Objets de données process (PDO) pour des données en temps réel,
- Objets de données service (SDO) pour la transmission de paramètres et de programmes,
- Gestion du réseau (NMT, Life-Guarding, Heartbeat)
- Objets prédéfinis (pour la synchronisation, l'horodatage, les cas d'urgence)

Tous les paramètres de l'appareil sont sauvegardés dans un **répertoire d'objets**. Ce répertoire d'objets contient la description, le type de donnée et la structure des paramètres, ainsi que l'adresse (index).

Le répertoire se décompose en une partie profil de communication, une partie profil d'appareil et une partie spécifique au constructeur.

1.3 Profil d'appareil pour codeur DS 406 V3.1

Ce profil fournit une définition de l'interface pour les codeurs **indépendante des constructeurs** et obligatoire. Ce profil définit les fonctions CANopen à utiliser, ainsi que la manière de les utiliser. Cette norme permet la réalisation d'un système de bus ouvert et indépendant des constructeurs.

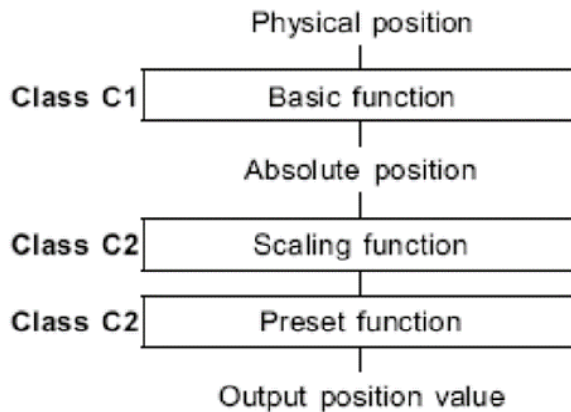


Figure 1

Le profil d'appareil se décompose en deux classes d'objets :

- La **Classe C1** décrit l'ensemble des fonctions de base dont le codeur doit être muni
- La **Classe C2** comporte de nombreuses fonctions étendues qui doivent être supportées par les codeurs de cette classe (Mandatory-obligatoires) ou qui sont en option. Les appareils de la classe 2 disposent ainsi de toutes les fonctions C1 et C2 obligatoires et d'autres fonctions optionnelles dépendant du constructeur. Le profil définit en outre une plage d'adresses pouvant être occupées par des fonctions spéciales spécifiques au constructeur.

1.4 Objectifs de LSS

Le *Layer Setting Service* et le protocole (LSS) CiA DSP 305 CANopen ont été développés pour permettre la lecture et la modification des paramètres suivants par l'intermédiaire du réseau :

1. ID du nœud CANopen
2. Vitesse de transmission CAN
3. Adresse LSS

Ces possibilités augmentent la compatibilité « Plug-and-Play » des appareils dans les réseaux CANopen, la configuration préalable du réseau étant moins restrictive. Le maître LSS est en charge de la configuration de ces paramètres pour un ou plusieurs esclaves LSS du réseau CANopen.

1.5 Transmission de données

CANopen fait appel, pour la transmission des données, à deux types de communication différents (COB=Communication Object) avec des caractéristiques différentes :

- **Objets de données process (PDO – avec possibilité de fonctionnement en temps réel)**
- **Objets de données service (SDO)**

Les objets de données process (**PDO**) sont utilisés pour l'échange hautement dynamique de données en temps réel (p. ex. position du codeur, vitesse, état des positions de référence) d'une longueur maximale de 8 octets. Ces données sont transmises avec une priorité haute (identifiant COB bas). Les PDO sont des données de diffusion en temps réel disponibles en même temps à tous les destinataires désirés. Les PDO peuvent être mappés : un mot de données de 8 octets peut comporter 4 octets pour la position et 2 octets pour la vitesse.

Les objets de données service (**SDO**) constituent le canal de communication pour la transmission de paramètres des appareils (p. ex. programmation de la résolution du codeur). Comme ces paramètres sont transmis de manière acyclique (p. ex. une seule fois lors du démarrage du réseau), les objets SDO ont une priorité basse (identifiant COB élevé).

1.6 Objets et codes de fonctions dans le Predefined Connection Set

Afin de faciliter la gestion des identifiants, CANopen utilise le « Predefined Master/Slave Connection Set » (télégramme maître/esclave prédéfini). Tous les identifiants sont définis ici dans le répertoire d'objets par des valeurs standard. Ces identifiants peuvent cependant être modifiés de manière spécifique au client par accès SDO.

L'identifiant à 11 bits se compose d'un **code de fonction à 4 bits** et d'un **numéro de nœud à 7 bits**. Plus la valeur de l'identifiant COB est élevée, plus sa priorité est basse !

Objets Broadcast (diffusion sur l'ensemble du réseau)

object	function code (binary)	resulting COB-ID	Communication Parameters at Index
NMT	000	0	-
SYNC	0001	128 (50h)	1005h, 1006h, 1007h
TIME STAMP	0010	256 (100h)	1012h, 1013h

Figure 2

Objets Peer-To Peer (appareil à appareil)

object	function cond (binary)	Resulting COB-IDs	Communication Parameters at Index
EMERGENCY	0001	129 (81h) – 255 (FFh)	1014h, 1015h
PDO1 (tx)	0011	385 (181h) – 511 (1FFh)	1800h
PDO1 (rx)	0100	513 (201h) – 639 (27Fh)	1400h
PDO2 (tx)	0101	641 (281h) – 767 (2FFh)	1801h
PDO2 (rx)	0110	769 (301h) – 895 (37Fh)	1401h
PDO3 (tx)	0111	897 (381h) – 1023 (3FFh)	1802h
PDO3 (rx)	1000	1025 (401h) – 1151 (47Fh)	1402h
PDO4 (tx)	1001	1153 (481h) – 1279 (4FFh)	1803h
PDO4 (rx)	1010	1281 (501h) – 1407 (57Fh)	1403h
SDO (tx)	1011	1409 (581h) – 1535 (5FFh)	1200h
SDO (rx)	1100	1537 (601h) – 1663 (67Fh)	1200h
NMT Error Control	1110	1793 (701h) – 1919 (77Fh)	1016h, 1017h

Figure 3

Objets d'utilisation restreinte et réservés

COB-ID	used by object
0 (000h)	NMT
1 (001h)	reserved
257 (101h) – 384 (180h)	reserved
1409 (581h) – 1535 (5FFh)	default SDO (tx)
1537 (601h) – 1663 (67Fh)	default SDO (rx)
1760 (6E0h)	reserved
1793 (701h) – 1919 (77Fh)	NMT Error Control
2020 (780h) – 2047 (7FFh)	reserved

Figure 4

1.7 Transmission des données process

Le codeur CANopen dispose de **trois services PDO** PDO1 (tx) ,PDO2 (tx) et PDO3(tx), et d'un **PDO de réception**. Une transmission PDO peut être déclenchée par différents événements (voir Répertoire d'objets Index 1800h) :

- **asynchrone** (géré par des événements), commandée par un temporisateur cyclique interne de l'appareil ou par un changement de valeurs process des données du capteur
- **synchrone** en réponse à un télégramme SYNC ; (l'instruction SYNC donne l'ordre à tous les nœuds CANopen de sauvegarder leurs valeurs de manière synchrone,
- afin de les envoyer ensuite successivement sur le bus conformément aux priorités définies),
- en **réponse** à un télégramme RTR (le Remote Frame=bit RTR récessif permet de demander exactement le message correspondant à l'identifiant transmis)

Les **messages PDO peuvent** présenter la structure suivante :

Process Data in Binary Code							
Byte 0 $2^7 \dots 2^0$	Byte 1 $2^{15} \dots 2^8$	Byte 2 $2^{23} \dots 2^{16}$	Byte 3 $2^{31} \dots 2^{24}$	Byte 4	Byte 5	Byte 6	Byte 7
PDO 3				Position value			
PDO 1				Position value			
PDO 2				Speed ²⁾		Acceleration ³⁾	

Figure 5

1Flags Octet d'état de l'objet Working area (zone de travail) 6400h

2Speed Mot de 16 bits signé

3Acceleration Mot de 16 bits signé

Le **PDO d'émission 1** est constitué (mappé) à partir de l'état de la valeur de position à 32 bits des **registres de la zone de travail** (6400h).

Le **PDO d'émission 2** est constitué à partir des valeurs de position (32 bits), de vitesse (16 bits) et d'accélération (16 bits).

Le **PDO d'émission 3** est un **PDO SYNC** de la position.

Toutes les **autres combinaisons de PDO** avec d'autres objets sont possibles, à la condition que la longueur de données maximale de 8 octets ne soit pas dépassée.

2. Installation électrique et bus CAN

2.1 Tension d'alimentation (alimentation électrique)

Relier la tension d'alimentation aux fils 1 et 2 (0V) et (+ VDC).

Type:		mA		VDC											
SIG.	0V	+V	CAN_H	CAN_L	CAN GND	CAN_H	CAN_L	CAN GND							
Lead nr.	1	2	4	5	6	7	8	9							PH
Ex	II 2G Ex d II C T6 PTB 09 ATEX 1106 X II 2D EX tD A21 IP6X T85°C CE 0102										Working temperature -40°...+60°C				

Figure 6

2.2 Lignes CANbus

Relier les lignes d'entrée CANbus aux fils 4 et 5 (CAN_H) et (CAN_L) et les lignes de sortie aux fils 7 et 8.

Type:		mA		VDC											
SIG.	0V	+V	CAN_H	CAN_L	CAN GND	CAN_H	CAN_L	CAN GND							
Lead nr.	1	2	4	5	6	7	8	9							PH
Ex	II 2G Ex d II C T6 PTB 09 ATEX 1106 X II 2D EX tD A21 IP6X T85°C CE 0102										Working temperature -40°...+60°C				

Figure 7

2.3 Terminaison du bus

Si l'appareil est la dernière station du bus, il faut terminer à ses deux extrémités le bus CAN bouclé au moyen d'une résistance terminale de bus placée entre CAN_H et CAN_L. Sur les appareils fermés, il faut spécifier l'activation ou non de la terminaison à la commande ; dans le cas contraire, la terminaison doit être réalisée en externe au moyen d'une résistance.

Relier la **résistance terminale (120 ohms)** aux fils **7 et 8 (CAN_H) et (CAN_L)**.

Type:		mA													
Series -Nr.		VDC													
SIG.	0V	+V	CAN_H	CAN_L	CAN GND	CAN_H	CAN_L	CAN GND							
Lead nr.	1	2	4	5	6	7	8	9							PH
Ex	II 2G Ex d II C T6 PTB 09 ATEX 1106 X II 2D EX tD A21 IP6X T85°C CE 0102										Working temperature -40°...+60°C				

Figure 8

3. Mise à la terre et liaison équipotentielle

Une mise à la terre et une liaison équipotentielle efficaces sont essentielles à la protection des réseaux CANbus contre les interférences. La mise à la terre et la liaison équipotentielle servent principalement à assurer le bon fonctionnement de CANbus et pas à assurer sa sécurité. Une bonne mise à la terre du blindage des câbles assure la réduction des interférences électrostatiques, minimisant ainsi les perturbations. La liaison équipotentielle assure un potentiel de terre identique sur l'ensemble du réseau. Ceci évite le passage de courants à la terre dans le blindage des câbles CANbus. Les informations ci-dessous donnent des indications d'ordre général pour la mise à la terre et la liaison équipotentielle.

Sur la station CANbus

Relier le **blindage du câble CANbus** à la liaison équipotentielle sur chaque station CANbus.

S'il est utilisé, le connecteur CANbus assure le raccordement du blindage du câble. Il faut cependant veiller dans ce cas à raccorder correctement le blindage dans le connecteur.

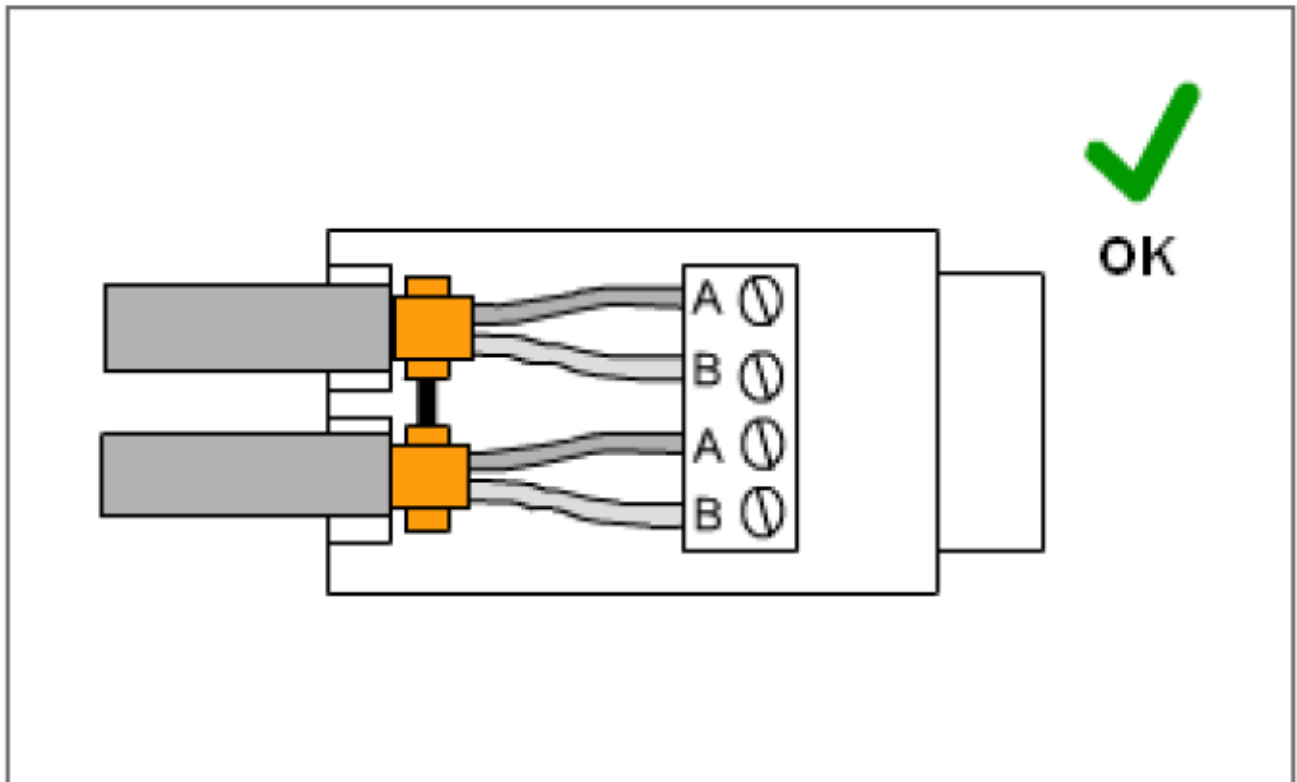


Figure 9

4. Guide de démarrage rapide - Réglages par défaut à la livraison

4.1 Codeurs avec départ de câble

Description	Réglage	Commutateur	Logiciel
Vitesse de transmission	250 Kbits/s	Réglage commutateur 5	Objet 2100h = 5h
Adresse de nœud	63	Réglage commutateur 3Fh	Objet 2100h = 3Fh
Terminaison	Off	Réglage commutateur off	Objet 2100h = 01h

Figure 10

4.2 Paramètres de communication

Index (hex)	Name	Standard value
1005h	COB-ID Sync	80h
100Ch	Guard Time	0
100Dh	Life Time Factor	0
1012h	COB-ID Time stamp	100h
1013h	High Resolution time stamp	0
1017h	Producer heartbeat time	0
1029h	Error Behaviour	0 = Comm Error
		1 = Device specific
		1 = Manufacturer Error
1800h	TPD01 Communication Parameter	
01h	COB-ID	180h + Node number
02h	Transmission Type	255 (asynch)
03h	Inhibit Time	0 [steps in 100µs]
05h	Event Timer	0 [steps in ms]
1801h	TPD02 Communication Parameter	
01h	COB-ID	280h + Node number
02h	Transmission Type	01 (asynch)
03h	Inhibit Time	0 [steps in 100µs]
05h	Event Timer	0 [steps in ms]
1802h	TPD03 Communication Parameter	

01h	COB-ID	380h + Node number
02h	Transmission Type	255 / 0xFFh (synch)
03h	Inhibit Time	0 [steps in 100µs]
05h	Event Timer	0 [steps in ms]
1A00h	TPD01 Mapping	
01h	1. Mapped Object	0x60040020
1A01h	TPD02 Mapping	
01h	2. Mapped Object	0x60300110
1A02h	TPD03 Mapping	
01h	3. Mapped Object	0x60040020

Figure 11

4.3 Profil Codeur

Index (hex)	Name	Standard value
6000h	Operating Parameter	0x04h Scaling on
6001h	Measuring Units per Revolution	8192 (13 Bit)
6002h	Total Measuring Range	33554432 (25 Bit)
6003h	Preset value	0
6200h	Cyclic Timer (see TPD01 Comm. Par)	0
6401h	Work area low limit	0
6402h	Work area high limit	65535
2105h	Save all Bus Parameters	0x65766173
2130h	Encoder Measuring Step	
	Speed Calculation Multiplier	10
	Speed Calculation Divisor	10
	Speed average value	10

Figure 12

Les valeurs standards d'origine (valeurs par défaut à la livraison) peuvent être rechargées à l'aide de l'objet **1011h** (restauration des paramètres). Pour sauvegarder les paramètres modifiés de sorte à les protéger contre les pannes de courant, il faut impérativement les transférer dans l'EEPROM à l'aide de l'objet **1010h** (sauvegarde des paramètres). Toutes les données sauvegardées auparavant dans l'EEPROM sont écrasées !

Si des erreurs sont survenues lors de la programmation des objets et si ces paramètres ont été sauvegardés dans l'EEPROM, le codeur ne répondra plus à la mise en service suivante (il n'émettra plus que des messages **Emergency**). Ce défaut ne peut être corrigé que par un **Reset** général du codeur.

Réglage par défaut du mappage du PDO d'émission :

Mapping	TPD01	TPD02	TPD03
1. Mapping	0x60040020	0x60300110	0x60040020
Object	6004h	6030h	6004h
Subindex	00	01	00
Data length	20h (32 Bit)	10h (16 Bit)	20h (32 Bit)
	Asynchron	Asynchron	Synchron

Figure 13

Le codeur CANopen supporte un **mappage variable** sur les 3 PDO d'émission.

5. Objets Emergency

Les objets Emergency apparaissent dans des situations d'erreurs dans un réseau CAN ; ils sont déclenchés en fonction des événements et émis sur le bus avec une **priorité haute**.

Important : un objet Emergency n'est déclenché qu'une fois par « évènement ». Aucun nouvel objet n'est généré tant que l'erreur est présente. Lorsque l'erreur est éliminée, un nouvel objet Emergency est généré avec le contenu 0 (Error Reset ou No Error) et émis sur le bus.

Codes d'erreur supportés

Les codes d'erreur sont indiqués en **orange**.

Error Code (hex)	Meaning
00xx	Error Reset or No Error
10xx	Generic Error
20xx	Current
21xx	Current, device input side
22xx	Current inside the device
23xx	Current, device output side
30xx	Voltage
31xx	Mains Voltage
32xx	Voltage inside the device
33xx	Output Voltage
40xx	Temperature
41xx	Ambient Temperature
42xx	Device Temperature
50xx	Device Hardware
60xx	Device Software
61xx	Internal Software
62xx	User Software
63xx	Data Set
70xx	Additional Modules
80xx	Monitoring
81xx	Communication
8110	CAN Overrun (Objects lost)
8120	CAN in Error Passive Mode
8130	Life Guard Error or Heartbeat Error
8140	recovered from bus off
8150	Transmit COB-ID collision
82xx	Protocol Error
8210	PDO not processed due to length error
8220	PDO length exceeded
90xx	External Error
F0xx	Additional Functions
FFxx	Device specific

Figure 14

6. Protocole Heartbeat

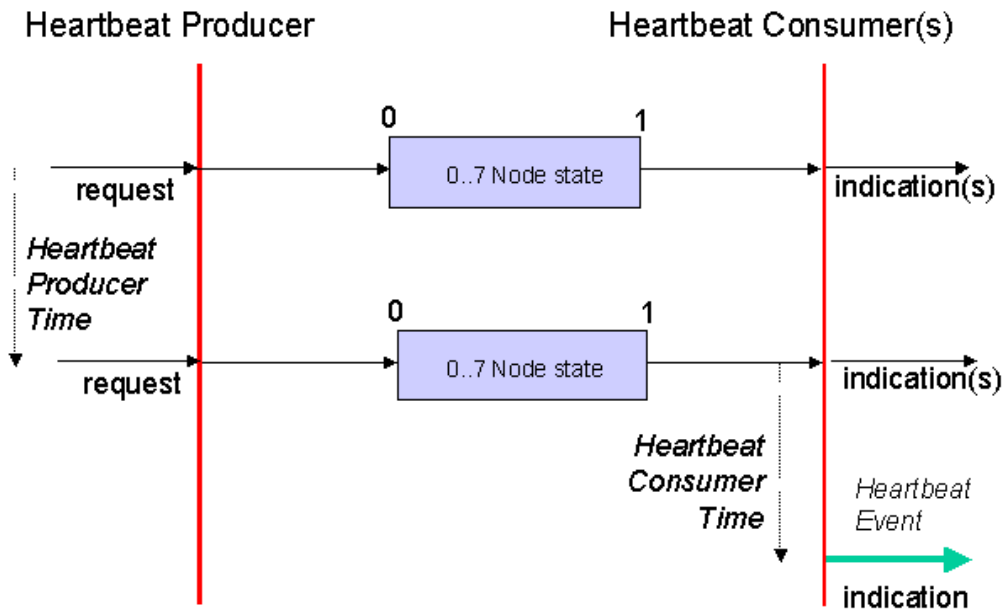


Figure 15

En alternative au **Node-Guarding**, il faut dorénavant utiliser le protocole moderne **Heartbeat**. Ce protocole est activé en définissant dans l'**objet 1017h** Producer Heartbeat Time une valeur > 0. Un « Heartbeat Producer » (émetteur Heartbeat) émet ce message Heartbeat de manière cyclique. Un ou plusieurs « Heartbeat Consumers » (récepteurs Heartbeat) peuvent recevoir ce message Heartbeat. Si l'émission cyclique de ce message Heartbeat disparaît, un « Heartbeat Event » se déclenche. Le comportement en cas de défaut est décrit dans l'objet 1029h sous-index 1 « Erreur de communication ».

7. Objets du Profil Codeur DS 306

7.1 Objet 6000h Paramètres de fonctionnement

- Bit 0 : Séquence du code : 0 = croissant pour une rotation en sens horaire (cw)
1 = croissant pour une rotation en sens antihoraire (ccw)
Défaut : Bit = 0
- Bit 2 : Fonction d'échelle : 0 = disable, 1 = enable ; Standard : Bit = 1 (voir Objet 6001,6002)
Défaut : Bit = 1
- Bit 13 : Format de la vitesse : 0 = trs/min, 1 = unités/seconde
Défaut : Bit = 0
- Bit 14 : Mode de démarrage : 0 = mode Pre-Operational après Bootup, 1 = mode Operational après Bootup
Défaut : Bit = 0
- Bit 15 : Mode Event : 0 = émission de la position selon TPDO 1800h, 1 = émission à chaque changement de position
Défaut : Bit = 0

Bit	Function	Bit = 0	Bit = 1	C1	C2
0	Code sequence	CW	CCW	M	M
1	Commissioning Diagnostic Control	Disabled	Enabled	O	O
2	Enable scaling	Disabled	Enabled	O	M
3	Measuring direction	Forward	Reverse	O	O
4 ... 11	Reserved for further use				
12	Manufacturer specific parameter	N.A.	N.A.	O	O
13	Speed Format	RPM	Unit/sec	O	O
14	Sartup automatic in OP-Mode	Disabled	Enabled	O	O
15	Event Mode Position*	Disabled	Enabled	O	O

Figure 16

* Régler ce mode à 254 dans Transmission Type dans les TPDO

M = la fonction doit être supportée

O = optionnel

7.2 Objet 6001h : nombre de pas de mesure par tour (Résolution)

Ce paramètre permet le réglage de la résolution requise par tour. Le codeur calcule alors en interne le facteur d'échelle correspondant. Le facteur d'échelle calculé MURF (par lequel la valeur de la position physique sera multipliée) se calcule selon la formule suivante :

MURF = Nombre de pas de mesure par tour (6001h) / résolution physique monotour (6501h)

Contenu des données :

Byte 0	Byte 1	Byte 2	Byte 3
$2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$	$2^{31} \dots 2^{24}$

Plage de valeurs : 1...résolution physique maximale (65536) 16 bits

Réglage par défaut : **8192 (13 bits)**

7.3 Objet 6002h : Nombre total de pas de mesure

Ce paramètre permet le réglage du nombre total de pas de mesure **monotour et multitours**. La résolution physique maximale est affectée d'un facteur. Ce facteur est toujours < 1 . Après avoir atteint le nombre total des pas de mesure, le codeur se remet à zéro.

Contenu des données :

Byte 0	Byte 1	Byte 2	Byte 3
$2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$	$2^{31} \dots 2^{24}$

Plage de valeurs : 1...résolution physique maximale (268435456) 28 bits

Réglage par défaut : **33554432 (25 bits)**

Exemple : Saisie 20000h

La valeur de la position physique sera multipliée par un facteur 0. XXXXXX et émise comme la position finale.

7.4 Objet 6003h : Valeur de prépositionnement

La valeur de la position du codeur est réglée à cette valeur de prépositionnement. Il est ainsi possible, par exemple, d'aligner la position zéro du codeur et la position zéro de la machine.

Contenu des données :

Byte 0	Byte 1	Byte 2	Byte 3
$2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$	$2^{31} \dots 2^{24}$

Plage de valeurs : 1...résolution physique maximale (268435456) 28 bits

Réglage par défaut : **0**

7.5 Objet 6004h : Valeur de position

Le codeur émet la valeur de la position courante (le cas échéant affectée du facteur d'échelle).

Contenu des données :

Byte 0	Byte 1	Byte 2	Byte 3
$2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$	$2^{31} \dots 2^{24}$

Plage de valeurs : 1...résolution physique maximale (268435456) 28 bits

7.6 Objet 6030h : Valeur de vitesse

Le codeur émet la vitesse calculée courante (le cas échéant affectée du facteur d'échelle) sous la forme d'une valeur en 16 bits.

La vitesse dépend des **réglages de l'objet 2130h**. Ces valeurs affectent le calcul et le résultat.

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Plage de valeurs : 0...vitesse maximale 15000 tours/min

AVIS	<p>Pour les valeurs supérieures à 12000 tours/min., un message d'avertissement est émis et le bit d'avertissement « Dépassement de la vitesse de rotation bit 0 » de l'objet Avertissements 6505h est activé.</p> <p>D'autres paramètres pouvant également affecter cet objet sont mentionnés dans 2130h.</p>
-------------	---

7.7 Objet 6040h : Valeur d'accélération

Le codeur émet la valeur calculée de l'accélération courante (avec le signe correspondant) sous la forme d'une valeur signée en 16 bits. L'accélération se calcule à partir des changements de vitesse et dépend donc indirectement des **réglages de l'objet 2130h**. Ces valeurs affectent le calcul et le résultat.

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

AVIS	<p>Plage de valeurs : 0.... +/- accélération maximale</p> <p>Une valeur négative indique une accélération négative (la vitesse diminue)</p>
-------------	---

Une accélération moyenne **a** correspond à l'évolution dans le temps de la vitesse **v** et peut ainsi se décrire de manière formelle par la dérivée de la vitesse en fonction du temps **t** ; une accélération **moyenne** est calculée ici sur la base de la différence des vitesses Δv à deux moments différents Δt ($t_2 - t_1$).

$$a = \Delta v / \Delta t \text{ ou } a = v_2 - v_1 / t_2 - t_1$$

7.8 Objet 6200h : Temporisateur cyclique

Définit le temps du cycle selon lequel la position courante est émise via PDO 1 (voir objet 1800h). Cette sortie commandée par temporisateur est activée dès qu'un temps de cycle > 0 est défini.

AVIS	Cet objet n'existe plus que pour des raisons de compatibilité avec les versions de profils plus anciennes. Utiliser le sous-index Event Timer (05h) dans le PDO d'émission courant à la place de cet objet.
-------------	---

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Plage de valeurs : 0 ... FFFFh (65535) donne le temps de cycle en millisecondes

Valeur standard = 0h

7.9 Objet 6500h : Affichage de l'état de fonctionnement

Cet objet indique l'état des réglages programmés pour l'objet 6000h.

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Contenu des données : voir l'objet 6000h

7.10 Objet 6502h : Nombre de tours multitours

Cet objet spécifie le nombre de tours que le codeur multitours doit compter. Cette valeur dépend du type de codeur ; toute valeur entre 4096 (12 bits) et 65535 (16 bits) est possible.

Cette valeur prédéfinie n'affecte que le nombre de tours. Elle n'affecte pas la résolution.

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Plage de valeurs : 4096 à 65535

Le réglage par défaut 1000h correspond à 4096.

7.11 Objet 6503h : Alarmes

En plus des erreurs signalées par les messages d'urgence (emergency), l'objet 6503h dispose d'autres messages d'erreur. Le bit d'erreur correspondant est mis à 1 tant que l'erreur est présente.

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Bit No.	Description	Value = 0	Value = 1
Bit 0	Position error	Position value valid	Position error
Bit 1	Hardware check	No error	Error
Bit 2...15	Not used		

Figure 17

Dans les deux cas, un message d'urgence (**ID=80h+numéro de nœud**) est envoyé en même temps que le code d'erreur **1000h (Erreur générique)** lors de l'apparition d'une alarme.

7.12 Objet 6504h : Alarmes supportées

Cet objet indique les messages d'alarme supportés par le codeur (voir objet 6503h).

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Plage de valeurs : voir l'objet 6503h

Un bit à 1 signifie que le message d'alarme est supporté

Exemple : Bit 0 = 1 L'affichage du défaut de position est supporté

7.13 Objet 6505h : Avertissements

Les messages d'avertissement signalent le dépassement des tolérances de paramètres internes du codeur. A l'opposé des messages d'alarme ou d'urgence, la valeur de la mesure peut être valide malgré tout lors de l'apparition d'un message d'avertissement. Le bit d'avertissement correspondant est mis à 1 tant que le dépassement de la tolérance ou l'avertissement est présent.

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Bit No.	Description	Value = 0	Value = 1
Bit 0	Overspeed	none	exceeded
Bit 1	Not used		
Bit 2	Watchdog Status	System OK	Reset carried out
Bit 3	Operting time	Below < 100000h	> 100000h
Bit 4...15	Not used		

Figure 18

Si le bit 2 ou le bit 3 est activé, un message Emergency (ID=80h+numéro de nœud) est émis en même temps que le **code d'erreur 5200h** (Hardware appareil).

7.14 Objet 6506h : Avertissements supportés

Cet objet indique les messages d'avertissement supportés par le codeur (voir objet 6505h).

Contenu des données :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Plage de valeurs : voir l'objet 6505h

Un bit à 1 signifie que le message d'avertissement est supporté.

7.15 Objet 6400h : Registre d'état de la zone de travail 2 valeurs

Cet objet contient l'état actuel de la position du codeur par rapport aux limites programmées. Les drapeaux sont activés ou désactivés en fonction de la position des deux valeurs limites. Cette comparaison avec les deux valeurs limites s'effectue en « temps réel » et peut s'utiliser pour le positionnement en temps réel ou pour le déclenchement en fin de course.

Work_area_state							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1 = CCW		smaller than LowLimit2	larger than HighLimit2	inside range2	smaller than LowLimit1	larger than HighLimit 1	inside range1

Figure 19

Plage de valeurs 8 bits

Contenu des données voir bit 0...7

AVIS	Il faut vérifier les deux valeurs limites objet 6401h et objet 6402h afin de s'assurer que les signaux de sortie s'activent correctement.
-------------	---

Objet 6401h : Limite inférieure zone de travail 2 valeurs

Objet 6402h : Limite supérieure zone de travail 2 valeurs

Ces deux paramètres permettent de définir la zone de travail. L'état, à l'intérieur ou à l'extérieur de cette zone, peut être signalé au moyen d'octets drapeaux (**objet 6400h Etat de la zone de travail**). Ces marqueurs de zone peuvent également faire office de fins de course logiciels.

Contenu des données :

Byte 0	Byte 1	Byte 2	Byte 3
$2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$	$2^{31} \dots 2^{24}$

Plage de valeurs : 1...résolution physique maximale (268435456) 28 bits

Réglage par défaut : 33554432 (25 bits) Limite supérieure zone de travail
0 Limite inférieure zone de travail

7.16 Objet 2100h : Vitesse de transmission

Cet objet permet de modifier la vitesse de transmission par le logiciel. Le réglage par défaut est FFh, indiquant que le réglage de la vitesse de transmission par hardware est prioritaire. Si la valeur est réglée entre 1..9 et si ce paramètre est sauvegardé, l'appareil utilisera la vitesse de transmission modifiée au redémarrage ou au Reset Node suivant. Après la modification de la vitesse de transmission, les paramètres doivent être sauvegardés de manière permanente dans l'EEPROM à l'aide de l'**objet 2105h**.

Contenu des données :

Byte 0
$2^7 \dots 2^0$

Plage de valeurs 1... 8 (voir le tableau Sélecteurs hardware : Vitesse de transmission CANopen)

Réglage par défaut : 0x05

7.17 Objet 2101h : Adresse de nœud

Cet objet permet de modifier l'adresse de nœud par le logiciel. Le réglage par défaut est 0xFFh, indiquant que le réglage de l'adresse de nœud par hardware est prioritaire. Après la modification de l'adresse de nœud, les paramètres doivent être **sauvegardés de manière permanente dans l'EEPROM à l'aide de l'objet 2105h**.

Contenu des données :

Byte 0
$2^7 \dots 2^0$

Plage de valeurs : 1 ...127 ou 1..7Fh

Réglage par défaut : FFh

Le **numéro de nœud** 0 est réservé et ne doit pas être utilisé pour définir un nœud. Les numéros de nœud résultants se trouvent dans la plage **1...7Fh** hexadécimal ou (1...127).

Un nouveau numéro de nœud n'est pris en compte qu'au redémarrage suivant (Reset/Power-on) du codeur ou au moyen d'une instruction **NMT Reset Node**. Tous les autres réglages dans la table d'objets sont conservés.

7.18 Objet 2102h : Terminaison de bus CAN OFF/ON

Cet objet permet de d'activer la terminaison de bus par le logiciel. Par défaut, cette valeur est réglée à 0, ce qui signifie que le réglage de la terminaison bus par hardware est prioritaire. Après la modification de l'état de la terminaison de bus, les paramètres doivent être sauvegardés de manière permanente dans l'EEPROM à l'aide de l'**objet 2105h**.

Contenu des données :

Byte 0
$2^7 \dots 2^0$

Plage de valeurs 0 ... 1

Réglage par défaut : 0 *Terminaison activée pour les codeurs avec départ de câble et un connecteur M12.

Il faut noter que, si l'activation de la terminaison par logiciel est sélectionnée, l'activation par hardware ne fonctionne pas et vice-versa.

7.19 Objet 2103h : Version du firmware flash

Cet objet indique la version actuelle du firmware sous la forme d'une valeur hexadécimale en 16 bits. Cette valeur permet de vérifier le niveau courant de l'appareil.

Contenu des données :

Byte 0
$2^7 \dots 2^0$

Plage de valeurs : jusqu'à FFFFh

Exemple : 4FA6h Firmware actuel

7.20 Objet 2105h : Save All Bus Parameters (enregistrement de tous les paramètres du bus)

Ce paramètre sauvegarde tous les paramètres du bus (objet 2100h, 2101h, 2102h) de manière permanente dans l'EEPROM. Utiliser l'instruction « save » (enregistrer tous les paramètres) pour enregistrer tous les paramètres. Cette opération nécessite env. 200 ms. Fin d'éviter tout enregistrement involontaire, l'instruction ne sera exécutée que si la chaîne de caractères « save » est saisie comme mot de code dans ce sous-index.

Un accès en lecture à cet index renvoie 0xFFFFFFFF.

Byte 0	Byte 1	Byte 2	Byte 3
$2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$	$2^{31} \dots 2^{24}$

Octet 3 : 73h (code ASCII pour « s »)

Octet 2 : 61h (code ASCII pour « a »)

Octet 1 : 76h (code ASCII pour « v »)

Octet 0 : 65h (code ASCII pour « e »)

Plage de valeurs : « save » en hexadécimal **0x65766173**

7.21 Objet 2110h : Données de configuration du capteur

Cet objet renvoie des informations sur la configuration du capteur. Les valeurs d'usine par défaut sont retournées par défaut ; aucune modification n'est nécessaire ici en temps normal.

Byte 0	Byte 1	Byte 2
$2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$

Plage de valeurs : 0... FF,FFh...

7.22 Objet 2120,4h : Température courante capteur de position *

Cet objet indique la température courante. La température courante à l'intérieur du capteur est mémorisée toutes les 6 minutes dans l'**objet 2150,sub1 Dernière température mémorisée**. Les températures maximale et minimale sont mémorisées dans l'

objet 2130, sub2 et sub3. Plage maximale de valeurs : 1...256.

Byte 0
$2^7 \dots 2^0$

Plage de valeurs 00...FFh

Exemple : **0x59** correspond environ à **25°C**

Les indications de température suivantes peuvent être prises comme référence :

-20°C correspond à **0x2Ch**

0°C correspond à **0x40h**

100°C correspond à **0xA4h**

Exemple : Valeur sélectionnée **0x71h de l'objet 2120,4h**

$0x71h - 0x40h = 0x31h$ correspond à 49°C en décimal

AVIS

Cet objet peut être mappé dans l'information PDO. La précision moyenne de la mesure est de $\pm 6^\circ\text{C}$, la mesure est réalisée à l'intérieur de l'électronique du codeur.

7.23 Objet 2120,2h : Température courante limite inférieure- Capteur de position

7.24 Objet 2120,3h : Température courante limite supérieure- Capteur de position

Ces deux paramètres permettent de définir les limites de la plage de température. Une température hors de cette plage est signalée par un **message Emergency**. Ces marqueurs de zone peuvent également faire office de détecteurs de limite de température.

Byte 0
$2^7 \dots 2^0$

Plage de valeurs 00...FFh

Exemple : **0x20** correspond à env. **-32°C**

Les indications de température suivantes peuvent être prises comme référence :

-20°C correspond à **0x2Ch**

0°C correspond à **0x40h**

100°C correspond à **0xA4h**

Plage de valeurs : **0x20h .. 0xACh**

Réglages par défaut : **0xA2h** Limite supérieure de la température

0x20h Limite inférieure de la température

7.25 Objet 2130h : Pas de mesure du codeur

Le paramètre **objet 2130, sub1 Multiplicateur de calcul de la vitesse** permet, par exemple, de définir la circonférence d'une roue mesureuse, afin de fournir la position en mm.

Cet objet influence la sortie de la vitesse. L'**objet 2130, sub2 Diviseur de calcul de la vitesse** permet de définir un diviseur pour un facteur d'unité. Définir dans l'**objet 2130, sub3 Valeur pour le calcul de la moyenne de la vitesse** le nombre de valeurs mesurées à utiliser pour le calcul de la moyenne glissante. Plage maximale de valeurs : 1...32.

Ces paramètres n'affectent que les **unités par seconde**.

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Plage de valeurs : voir le tableau

2130h Sub 1 Speed Calculation Multiplier, réglage par défaut : 10

2130h Sub 2 Speed Calculation Divisor, réglage par défaut : 10

2130h Sub 3 Speed Average Calc, réglage par défaut : 10

7.26 Objet 2140h : Mémoire utilisateur (16 octets)

Ces 4 paramètres constituent une zone mémoire destinée à l'utilisateur. **Il est possible de mémoriser 4 mots de données de 4 octets au maximum.** Le contenu de cette zone n'est pas vérifié, ce qui permet d'enregistrer n'importe quel format.

Contenu des données :

Byte 0	Byte 1	Byte 2	Byte 3
$2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$	$2^{31} \dots 2^{24}$

Plage de valeurs : numériques, alphanumériques

Réglage par défaut : 0

7.27 Objet 2150h : Historique de la température

Cet objet permet de consulter la température. La température courante à l'intérieur du capteur est mémorisée toutes les 6 minutes dans l'**objet 2150, sub1 Dernière température mémorisée**. Les températures maximale et minimale sont mémorisées dans l'**objet 2130, sub2 et sub 3**. Plage maximale de valeurs : 1...256.

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Contenu des données :

Une valeur de 0x50 correspond à env. 20°C

Une valeur de 0x40 correspond à env. 0°C

Une valeur de 0x90 correspond à env. 85°C

2150h Sub 1 Dernière température mémorisée	Réglage par défaut : 0x50
2150h Sub 2 Température maximale	Réglage par défaut : 0x50
2150h Sub 3 Temperature minimale	Réglage par défaut : 0x50
2150h Sub 4 Octet drapeau	Réglage par défaut : 0

Objet 1029h : Comportement en cas de défaut

Lorsqu'une erreur grave est détectée, l'appareil doit passer automatiquement dans le mode **Pre-Operational**. Cet objet permet de déterminer le comportement de l'appareil lors de l'apparition d'une erreur. Les classes d'erreurs suivantes sont couvertes :

1029h,sous-index 1 Erreurs de communication

- Etat Bus off de l'interface CAN
- Apparition d'un événement Life guarding
- Echec de la surveillance du Heartbeat

1029h,sous-index 2 Spécifique au profil de l'appareil

- Erreurs du capteur et du contrôleur
- Erreur de température

1029h,sous-index 3 Spécifique au profil du fabricant

- Erreur du contrôleur interne

La valeur des classes d'erreurs se décompose comme suit :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

Plage de valeurs : 8 bits

- 0 Mode Pre-Operational (uniquement si le mode Operational était actif auparavant)
- 1 Pas de changement de mode
- 2 Mode Stopped
- 3...127 réservés

Objets non mentionnés

Tous les objets non mentionnés ici fournissent des informations complémentaires et sont décrits dans le **profil codeur DS 406 V3.1**.

8. Gestion du réseau

Le codeur supporte la gestion de réseau simplifiée (minimum boot-up) définie dans le profil pour les « minimum capability devices ».

Le diagramme d'état selon DS 301 suivant représente les différents états des nœuds et les instructions réseau correspondantes (commandées par le maître du réseau à l'aide de services NMT).

- | | |
|------------------|---|
| (1) | Initialisation automatique de l'état NMT à la mise sous tension |
| (2) | Initialisation de l'état NMT terminée - passage automatique dans l'état NMT Pre-operational |
| (3) | Service NMT indication démarrage nœud distant ou par commande locale (démarrage autonome) |
| (4), (7) | Service NMT indication passage en pre-operational |
| (5), (8) | Service NMT indication arrêt nœud distant |
| (6) | Service NMT indication démarrage nœud distant |
| (9), (10), (11) | Service NMT indication réinitialisation nœud |
| (12), (13), (14) | Service NMT indication réinitialisation communication |

Initialisation :

Etat initial de l'appareil après sa mise sous tension, après un Reset ou après la mise en service de l'appareil.

Après l'exécution des routines de Reset/d'initialisation, le nœud passe automatiquement dans l'état Pre-operational. Les LED indiquent l'état instantané.

Pre-operational :

Il est maintenant possible de s'adresser au nœud CAN par des messages SDO ou des instructions NMT à l'aide de l'identifiant standard. La programmation des paramètres du codeur ou de la communication a lieu maintenant.

Operational :

Le nœud est actif. Les valeurs du process sont émises par l'intermédiaire des PDO. Toutes les instructions NMT peuvent être exploitées.

Prepared on Stopped :

Dans cet état, le nœud n'est plus actif : ni la communication SDO ni la communication PDO ne sont plus possibles. Le nœud peut être mis dans l'état Operational ou Pre-Operational à l'aide d'instructions NMT.

9. Instructions NMT

Instructions NMT

Toutes les instructions NMT sont transmises sous la forme d'objets NMT non confirmés. Grâce au modèle de communication Broadcast (diffusé sur tout le réseau), les instructions NMT sont reconnues par tous les périphériques.

Un objet NMT est structuré de la manière suivante :

Byte 0	Byte 1
$2^7 \dots 2^0$	$2^{15} \dots 2^8$

ID COB = 0

Octet 0 = octet de l'instruction

Octet 1 = numéro de nœud

AVIS

L'ID COB de l'objet NMT est toujours 0

Le nœud destinataire est défini par le numéro de nœud. Si le numéro de nœud est 0, l'instruction s'adresse à tous les nœuds.

Kommandobyte (hex)	Description
01h	Start_Remote_Node: Switch to Operational
02h	Stop_Remote_Node: Switch to Prepared
80h	Enter_Pre-Operational_State: Switch to Pre-operational
81h	Reset_Node: Reset node ¹
82h	Reset_Communication: Reset Communication ²

Figure 20

¹ A la mise sous tension, les valeurs sont attribuées à tous les paramètres du répertoire d'objet entier.

² A la mise sous tension, les valeurs sont attribuées uniquement aux paramètres de la section Profil de communication du répertoire d'objets.

10. Glossaire

Vitesse de transmission :

La vitesse de transmission est en relation avec la synchronisation nominale des bits. La vitesse de transmission maximale possible dépend de nombreux facteurs qui influencent le temps de propagation du signal sur le bus. Il existe une relation importante entre la vitesse de transmission maximale, la longueur du bus et le type de câble. Pour CANOpen, différentes vitesses de transmission peuvent être définies entre 10 kbit/s et 1 Mbit/s.

CANOpen :

CANOpen est un protocole basé sur CAN développé à l'origine pour les systèmes de contrôle industriels. Ses spécifications comprennent différents profils d'appareil, ainsi que le cadre pour des applications spécifiques. Les réseaux CANOpen s'utilisent également dans les véhicules tout-terrains, l'électronique marine, les équipements médicaux et les trains. La couche d'application très flexible et les nombreuses options en font le système idéal pour des solutions personnalisées. De nombreux outils de configuration sont en outre disponibles. L'utilisateur peut définir sur cette base des profils d'appareil spécifiques à son application. Davantage d'informations sur CANOpen sont disponibles sur l'Internet, à l'adresse www.can-cia.org.

Fichier EDS :

Le fichier EDS (Electronic Data Sheet - Fiche Technique Electronique) est fourni par le fabricant d'un appareil CANOpen. Son format standardisé permet la description des appareils. Le fichier EDS contient des informations sur :

- la description du fichier (nom, version, date de création, etc.)
- les informations générales de l'appareil (fabricant, nom et code)
- le nom et le type de l'appareil, la version, l'adresse LMT
- les vitesses de transmission supportées et son aptitude au démarrage
- la description par leurs attributs des objets supportés

Numéro de nœud :

Dans un réseau CANOpen, chaque appareil est défini par son numéro de nœud (ID de nœud). Les numéros de nœud autorisés se trouvent dans la plage de 0 à 127 et ne peuvent s'utiliser qu'une fois dans un réseau.



Kübler Group
Fritz Kübler GmbH
Schubertstr. 47
78054 Villingen-Schwenningen
Allemagne
Tél. : +49 7720 3903-0
Télécopie : +49 7720 21564
info@kuebler.com
www.kuebler.com