



Notice d'utilisation
Tête de lecture / écriture RFID
avec interface CANopen

FR

Contenu

1	Remarques préliminaires	4
1.1	Symboles utilisés	4
2	Consignes de sécurité	4
2.1	Remarques générales	4
2.2	Cible	4
2.3	Raccordement électrique	5
2.4	Interventions sur l'appareil	5
3	Informations générales	5
3.1	Technologie CANopen	5
3.2	Références	6
4	Fonctionnement et caractéristiques	6
5	Montage	6
5.1	Instructions de montage générales	6
5.2	Remarques sur le montage des tag	6
5.3	Élimination de perturbations	6
6	Éléments de visualisation des DTM424 / 425 / 428 / 434 / 435	7
7	Éléments de visualisation du DTC510	8
8	Interface CANopen	9
8.1	Fonctions CANopen	9
8.2	Modification du Node ID et du Bit-rate	10
8.2.1	Modification du Node ID et du Bit-rate dans le répertoire d'objets	10
8.2.2	Réglage du Node ID et du Bit-rate par LSS	10
8.3	Mise en service	11
8.4	Modes de communication des objets de données process (PDO)	12
8.5	Répertoire d'objets (RO)	13
8.6	Messages d'erreur	20
8.7	Surveillance d'activité avec Heartbeat	22
8.8	Modification d'objets	22
8.9	Objets de données process	23
8.9.1	Objets de données process émetteurs (TPDO)	23
8.9.2	Objets de données process récepteurs (RPDO)	24
8.10	État d'appareil	25
8.11	Désactiver l'antenne	27
8.12	Sélectionner le type de tag	27
8.13	Lecture des informations d'un tag	28
8.14	Valeur RSSI	28
8.15	Filtre de détection tag	29
8.15.1	Objet profondeur filtre UID	30
8.15.2	Objet profondeur filtre ID zéro	30
9	Transmission de données avec le tag	30
9.1	Lecture de l'UID (Unique Identification Number) du tag	30

9.2	Lecture de données du tag via transfert PDO	30
9.2.1	Exemple 1	31
9.2.2	Exemple 2	31
9.3	Ecriture de données sur le tag via transfert PDO	32
9.3.1	Exemple 1	33
9.3.2	Exemple 2	34
9.4	Traitement d'erreurs en cas de transfert PDO	34
9.5	Lecture des données du tag via transfert SDO	34
9.5.1	Exemple	35
9.6	Ecriture de données sur le tag via transfert SDO	35
9.6.1	Exemple	35
9.7	Blocage des zones de données sur le tag via transfert SDO	36
9.7.1	Exemple	36
9.8	Traitement d'erreurs en cas de transfert SDO	37
10	Données EDS	38
11	Maintenance, réparation et élimination	38
12	Glossaire	39

1 Remarques préliminaires

Ce document s'applique aux appareils du type "Tête de lecture / écriture RFID avec interface CANopen" (p.ex. référence : DTM425). Ce document fait partie de l'appareil.

Il s'adresse à des personnes compétentes. Ce sont des personnes qui sont capables – grâce à leur formation et expérience – d'envisager les risques et d'éviter des dangers potentiels qui pourraient être causés par le fonctionnement ou la maintenance de l'appareil. Le document fournit des informations sur l'utilisation correcte de l'appareil.

Lire ce document avant l'utilisation afin de vous familiariser avec les conditions environnantes, l'installation et le fonctionnement. Garder ce document pendant tout le temps d'utilisation de l'appareil.

Respecter les consignes de sécurité.

1.1 Symboles utilisés

▶ Action à faire

> Retour d'information, résultat

[...] Désignation d'une touche, d'un bouton ou d'un affichage

→ Référence croisée



Remarque importante

Le non-respect peut aboutir à des dysfonctionnements ou perturbations.



Information

Remarque supplémentaire.

2 Consignes de sécurité

2.1 Remarques générales

Cette notice fait partie de l'appareil. Elle fournit des textes et des figures pour l'utilisation correcte de l'appareil et doit être lue avant installation ou emploi.

Respecter les indications de cette notice. Le non-respect de ces consignes, une utilisation en dehors des conditions définies ci-dessous, une mauvaise installation ou utilisation peuvent avoir des conséquences graves pour la sécurité des personnes et des installations.

2.2 Cible

Cette notice s'adresse à des personnes considérées comme compétentes selon les directives CEM et basse tension. L'appareil doit être monté, raccordé et mis en service par un électricien habilité.

2.3 Raccordement électrique

Mettre l'appareil hors tension en prenant des mesures externes avant toutes manipulations.

Les broches de raccordement ne doivent être alimentées que par les tensions indiquées dans les données techniques et / ou sur l'étiquette de l'appareil et seuls les accessoires homologués d'ifm doivent être raccordés.



L'appareil ne dispose pas d'une résistance de terminaison CAN interne. Un câble de raccordement sans résistance de terminaison peut causer des perturbations sur le bus CAN.

- Utiliser des résistances de terminaison 120 Ω ou un câble de raccordement avec résistance de terminaison intégrée, par ex. l'article EVC492.

FR

2.4 Interventions sur l'appareil

En cas de mauvais fonctionnement de l'appareil ou en cas de doute prendre contact avec le fabricant. Des interventions sur l'appareil peuvent avoir des conséquences graves pour la sécurité des personnes et des installations. Elles ne sont pas autorisées et aboutissent à une exclusion de responsabilité et de garantie.

3 Informations générales

3.1 Technologie CANopen

Le profil de communication CANopen est basé sur la spécification CAN Application Layer (CAL) de l'organisation CiA (CAN in Automation). CANopen est considéré comme un bus de terrain robuste avec des options de configuration très flexibles. Il est utilisé dans de nombreuses applications différentes, qui toutes reposent sur des profils d'applications différents. CANopen contient un concept pour la configuration et la communication de données de temps réel, utilisant des messages synchrones et asynchrones. On distingue quatre types de messages (objets).

1. Messages d'administration (Layer Management, Network Management et Identifier Distribution)
2. Service Data Objects (SDO) / objets de données service
3. Process Data Objects (PDO) / objets de données process
4. Predefined Objects (Emergency) / objets prédéfinis

Pour plus d'informations voir la spécification CiA-CAN (CiA 301 - CANopen).

3.2 Références

<http://www.can-cia.org>

CAN Application Layer, DS 201...207	CiA
Profil LSS	DS305 CiA
Profil de communication basé sur CAN	DS 301 CiA
Spécification CAN version 2.0 A	Robert Bosch GmbH

4 Fonctionnement et caractéristiques

La tête de lecture / écriture RFID sert à la lecture et l'écriture de tag RFID. Le paramétrage et l'échange de données se font via l'interface CANopen.

Des applications typiques sont par exemple l'identification d'outils interchangeables et de pièces rapportées sur des engins mobiles.

5 Montage

5.1 Instructions de montage générales

-  Observer la notice de montage séparée.
-  En cas de montage de plusieurs systèmes respecter les distances minimales entre les têtes de lecture / écriture.
-  A proximité immédiate de sources d'émission HF, par ex. des transformateurs de soudure ou des convertisseurs, le fonctionnement des têtes de lecture / écriture peut être affecté considérablement.

5.2 Remarques sur le montage des tag

-  Le montage des tag en ou sur métal réduit la distance lecture et écriture.
-  L'orientation de l'axe de l'antenne de la tête de lecture / écriture doit correspondre à l'axe de la bobine du tag.

5.3 Elimination de perturbations

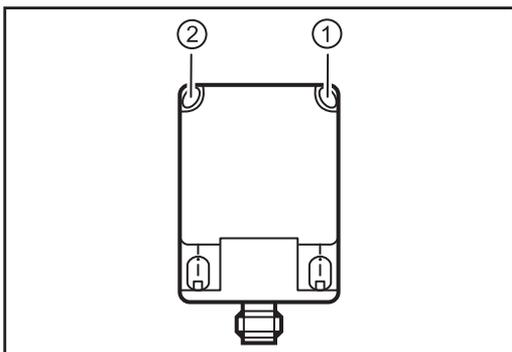
L'appareil génère un champ électrique modulé d'une fréquence de 13,56 kHz. Pour éviter de perturber la communication des données, il n'est pas permis de faire fonctionner d'autres appareils à proximité s'ils génèrent des émissions de rayonnements HF dans cette bande de fréquences, comme par exemple des variateurs de fréquence et des alimentations à découpage.

6 Eléments de visualisation des DTM424 / 425 / 428 / 434 / 435

Etat de fonctionnement	LED rouge	LED verte	LED jaune
Pre-operational	éteinte	allumée en permanence	éteinte
Pre-operational et tag détecté	éteinte	clignote en alternance avec LED jaune (toutes les 1,6 s)	clignote en alternance avec LED verte (toutes les 1,6 s)
Operational	éteinte	clignote (toutes les 0,4 s)	éteinte
Operational et tag détecté	éteinte	éteinte	allumée en permanence
Erreur de configuration	clignote (toutes les 0,4 s)	LED selon l'état de fonctionnement actuel	
Défaut dans le réseau CAN	clignote (toutes les 1,2 s)		
CAN : Bus Off	allumée en permanence	éteinte	éteinte
LSS Service actif	clignote	éteinte	éteinte
Défaut du matériel dans l'appareil détecté	éteinte	éteinte	clignote

FR

7 Eléments de visualisation du DTC510



- 1: verte (état de fonctionnement) / rouge (défaut)
2: jaune (tag)

Etat de fonctionnement	LED rouge	LED verte	LED jaune
Pre-operational	éteinte	allumée	éteinte ou allumée si un tag a été détecté dans le champ de lecture
Operational	éteinte	clignote (2,5 Hz)	éteinte ou allumée si un tag a été détecté dans le champ de lecture
Erreur de configuration	clignote en alternance avec LED verte (2,5 Hz)	clignote en alternance avec LED rouge (2,5 Hz)	éteinte ou allumée si un tag a été détecté dans le champ de lecture
Défaut dans le réseau CAN	clignote en alternance avec LED verte (0,8 Hz)	clignote en alternance avec LED rouge (0,8 Hz)	éteinte ou allumée si un tag a été détecté dans le champ de lecture
Bus CAN désactivé	allumée	éteinte	éteinte ou allumée si un tag a été détecté dans le champ de lecture
LSS Service actif	scintille irrégulièrement	éteinte	éteinte ou allumée si un tag a été détecté dans le champ de lecture
Défaut du matériel dans l'appareil détecté	éteinte	éteinte	scintille irrégulièrement

8 Interface CANopen

La tête de lecture / écriture RFID dispose d'une interface CANopen standardisée selon CiA DS-301. Toutes les valeurs mesurées et tous les paramètres sont accessibles via le répertoire d'objets (RO). La configuration individuelle peut être sauvegardée dans la mémoire permanente interne.

8.1 Fonctions CANopen

Les fonctions CANopen suivantes sont disponibles :

- 64 objets de données process émetteurs et récepteurs (TPDO1..64, RPDO1..64) en deux modes de fonctionnement possibles :
 - interrogation individuelle via un télégramme Remote-Transmit-Request (RTR)
 - transmission événementielle
- Messages d'erreur par objet d'urgence (EMCY) avec support :
 - du registre d'erreurs général
 - du registre d'état spécifique au fabricant
 - de la liste d'erreurs
- Mécanisme de surveillance Heartbeat
- Indication d'états et d'erreurs par LED
- Outre la fonctionnalité CiA DS-301, d'autres caractéristiques spécifiques aux fabricants et aux profils existent :
 - réglage du Node ID et du Bit-rate par une saisie dans le répertoire d'objets (SDO)
 - configuration et lecture / écriture de données de fonctionnement via objets de données service (SDO)
- Support de Layer Setting Service (LSS)
- Support de la transmission de données process synchronisée (SYNC)

8.2 Modification du Node ID et du Bit-rate

L'appareil offre plusieurs possibilités de modification du Node ID et du Bit-rate.

-  L'appareil est fourni avec le Node ID à 32 et un Bit-rate de 125 kbit/s.
-  Chaque Node ID ne doit être attribué qu'une seule fois dans le réseau CANopen. Si un Node ID est attribué plusieurs fois, des dysfonctionnements peuvent se produire dans le réseau CANopen.

8.2.1 Modification du Node ID et du Bit-rate dans le répertoire d'objets

Le Node ID est saisi dans le répertoire d'objets dans les objets 0x20F0 et 0x20F1. Si les deux valeurs se correspondent, le réglage est mémorisé et devient actif après réinitialisation du logiciel de l'appareil. Des valeurs entre 1 et 127 peuvent être utilisées comme Node ID.

Le Bit-rate est saisi dans les objets 0x20F2 et 0x20F3. Si les deux valeurs se correspondent, le réglage est mémorisé et devient actif après réinitialisation du logiciel de l'appareil. Les valeurs suivantes peuvent être utilisées comme Bit-rate :

Valeur	Bit-rate
0	1 000 kbits/s
1	800 kbits/s
2	500 kbits/s
3	250 kbits/s
4	125 kbits/s
5	100 kbits/s
6	50 kbits/s
7	20 kbits/s

-  Si un maître est utilisé dans le réseau CANopen pour la sauvegarde centralisée de paramètres, les valeurs modifiées pour Node ID (0x20F0 et 0x20F1) et Bit-rate (0x20F2 et 0x20F3) doivent également être saisies dans le maître.

Sinon, les valeurs seront remises à chaque démarrage du réseau CANopen.

8.2.2 Réglage du Node ID et du Bit-rate par LSS

A l'aide de Layer Setting Service (LSS), un maître LSS peut modifier le Node ID et le Bit-rate de l'appareil (esclave LSS) via le bus CAN. Pour ce faire, le maître LSS met tous les esclaves LSS en mode de configuration. Chaque esclave LSS peut être identifié sans équivoque via les données d'appareil (ID fournisseur, code produit, numéro de révision et numéro de série).

Pour changer le Bit-rate, le maître LSS transmet le nouveau Bit-rate en mode de configuration avec le service " Configure Timing Bit ". L'esclave LSS signale au maître LSS si le nouveau Bit-rate est supporté. Ensuite, le maître LSS transmet via le service "Activate Bit Timing" le temps "switch_delay" après lequel le nouveau Bit-rate doit être activé. Après l'activation, le maître LSS remet l'esclave LSS en mode opérationnel.

Pour modifier le Node ID, le maître LSS transmet le nouveau Node ID en mode de configuration. L'esclave LSS signale au maître LSS si le nouveau Node ID est valable. Après la modification du Node ID le maître LSS remet l'esclave LSS en mode opérationnel.

Le nouveau Bit-rate et le nouveau Node ID deviennent actifs après la réinitialisation du logiciel de l'esclave LSS.

FR

8.3 Mise en service

La norme CANopen CiA301 définit trois états de fonctionnement possibles pour les nœuds capteurs :

Pre-Operational

A l'état "Pre-Operational", aucun message PDO (données process) peut être transmis. L'état "Pre-Operational" s'utilise pour le paramétrage du capteur ou comme mode de veille.

Lors du démarrage, l'appareil répond en mode Pre-Operational sur le bus CAN avec le message Boot-up "0x700+Node ID".

Operational

A l'état "Operational" tous les services de communication sont effectués. L'état "Operational" s'utilise pour l'échange des données process pendant le fonctionnement.

Stopped

A l'état "Stopped" seulement des messages NMT (gestion du réseau) sont possibles. Ainsi, des capteurs redondants ou défectueux peuvent être séparés presque complètement du bus.

Le maître ou le gestionnaire du réseau peut demander au capteur de changer d'état via des messages NMT.

8.4 Modes de communication des objets de données process (PDO)

Le TPDO peut être vérifié à tout moment par transmission d'un message Remote Transmit Request (RTR). Sinon, les TPDO seront transmis automatiquement dès que leur valeur est changée (événementiellement).

En option, le service CANopen "SYNC" peut être utilisé (voir CiA 301, 7.2.5 Synchronization object (SYNC)). Pour la transmission synchronisée, CANopen met à disposition l'objet SYNC avec lequel les TPDO sont envoyés après chaque "Nième" réception d'un télégramme SYNC.

En total, 64 TPDO et 64 RPDO sont disponibles ; à la livraison seulement les 4 premiers sont actifs. Si la configuration du réseau CAN le permet, les objets de données process restants peuvent également être activés.

Les données process sont attribuées à la plage d'adressage linéaire du tag RFID dans les réglages de défaut. Le TPDO1 par exemple transmet les 8 premiers octets de la mémoire de données utilisables du tag RFID.

La lecture de la mémoire et la transmission des données via TPDO se font automatiquement dès qu'un nouveau tag RFID est détecté.

L'écriture des données sur les tag se fait de la même façon par un accès en écriture sur le RPDO correspondant.

 La transmission des données via objets de données process est seulement possible en mode de fonctionnement "Operational" (→ 8.3 Mise en service).

8.5 Répertoire d'objets (RO)

Index	Sub-index	Nom (objet)	Type	Accès	Valeur par défaut	Capacité mapping PDO	Sauvegarder valeur de l'objet
Communication CANopen (CiA 301)							
0x1000	0x00	Type d'appareil	u32	ro	0x00000000	-	-
0x1001	0x00	Registre d'erreurs	u8	ro	0x00	-	-
0x1003	0x01 0x02	Champ d'erreurs prédéfini	032	ro	0x00000000		
0x1005	0x00	COB ID SYNC	u32	rw	0x00000000	-	oui
0x1008	0x00	Nom de l'appareil fabricant	vSTR	ro	Numéro d'article de l'appareil	-	-
0x1009	0x00	Version du matériel fabricant	vSTR	ro	Version du firmware actuelle	-	-
0x100A	0x00	Version du logiciel fabricant	vSTR	ro	Version du logiciel actuelle	-	-
0x1010	0x01	Sauvegarder les paramètres (sauvegarder les paramètres appareil dans la mémoire non volatile)	u32	rw	0x00000000	-	-
0x1011	0x01	Charger les paramètres de communication par défaut	u32	rw	0x00000000	-	-
0x1014	0x00	COB ID EMCY (message de sécurité COB ID)	u32	rw	NodeID+ 0x80	-	
0x1015	0x00	Temps d'inhibition EMCY (temps d'inhibition entre des messages EMCY)	u16	rw	0x0000	-	oui
0x1017	0x00	Producteur temps heartbeat (différence temporelle entre des heartbeat transmis en ms)	u16	rw	0x0000	-	oui

FR

Index	Sub-index	Nom (objet)	Type	Accès	Valeur par défaut	Capacité mapping PDO	Sauvegarder valeur de l'objet
0x1018	0x01	ID fournisseur	u32	ro	0x0069666D	-	-
	0x02	Code produit	u32	ro	Code produit de la version appareil	-	-
	0x03	Numéro de révision	u32	ro	Révision principale et version logiciel actuelle	-	-
	0x04	Numéro de série	u32	ro	Numéro de série de l'appareil	-	-
0x1200	0x01	COB ID client au serveur	u32	ro	NodeID+ 0x600	-	-
	0x02	COB ID client au serveur	u32	ro	NodeID+ 0x580	-	-
0x1400-0x143F	0x01	Paramètres RPDO : COB ID	u32	rw	(→ 8.9.2)	-	oui
	0x02	Paramètres RPDO : type de transmission	u8	ro	0xFF	-	oui
0x1600-0x163F	0x01-0x08	Mapping RPDO	u32	rw	(→ 8.9.2)	-	oui
0x1800-0x183F	0x01	Paramètres TPDO : COB ID	u32	rw	(→ 8.9.1)	-	oui
	0x02	Paramètres TPDO : type de transmission	u8	ro	0xFF	-	oui
	0x03	Paramètres TPDO : temps d'inhibition	u16	rw	0x00	-	oui
0x1A00-0x1A3F	0x01-0x08	Mapping TPDO	u32	rw	(→ 8.9.1)	-	oui
Configuration bus							
0x20F0	0x00	Réglage NODE ID A (Node ID pour la communication CANopen)	u8	rw	32	-	Sauvegardé automatiquement
0x20F1	0x00	Réglage NODE ID B (Node ID pour la communication CANopen)	u8	rw	32	-	Sauvegardé automatiquement
0x20F2	0x00	Réglage Bit-rate A (Bit-rate Bus CAN)	u8	rw	4	-	Sauvegardé automatiquement

Index	Sub-index	Nom (objet)	Type	Accès	Valeur par défaut	Capacité mapping PDO	Sauvegarder valeur de l'objet
0x20F3	0x00	Réglage Bit-rate B (Bit-rate Bus CAN)	u8	rw	4	-	Sauvegardé automatiquement
Etat et commande appareil lecture							
0x2150	0x00	Etat d'appareil (bit état d'appareil)	u32	ro		oui	-
0x2151	0x00	Antenne active (frontend HF de l'appareil actif)	bool	rw	1	-	oui
0x2160	0x01-0xFE	Définition du type de tag (nom des tag supportés)	dom	ro	(→ 8.12)	-	-
0x2161	0x00	Sélection du type de tag (valeur sélectionne le type du tag qui est défini dans 0x2160)	u8	rw	2	-	oui
0x2162	0x00	RSSI	u8	ro	-	oui	-

Index	Sub-index	Nom (objet)	Type	Accès	Valeur par défaut	Capacité mapping PDO	Sauvegarder valeur de l'objet
Information tag							
0x2180	0x00	UID actuel (UID du tag qui se trouve dans la plage de lecture, PDO mappable)	u64	ro	0x00000000 00000000	oui	-
0x2181	0x00	DSFID actuel (DSFID du tag qui se trouve dans la plage de lecture, PDO mappable)	u8	ro	0x00	oui	-
0x2182	0x01	Information tag : UID	u64	ro	0x00000000 00000000	-	-
	0x02	Information tag : DSFID	u8	ro	0x00	-	-
	0x03	Information tag : AFI	u8	ro	0x00	-	-
	0x04	Information tag : taille mémoire	u32	ro	0x00000000	-	-
	0x05	Information tag : référence IC	u8	ro	0x00	-	-
	0x06	Information tag : type du tag (type du tag identifié, défini dans 0x2160)	u8	ro	0x00	-	-
Lecture des données mappables							
0x2200	0x01-0x40	Début lecture de l'adresse (début de la zone d'adresse sur le tag qui doit être lue)	u16	rw	(→ 8.9.2)	-	oui
0x2201	0x01-0x40	Lecture longueur (longueur de la zone de mémoire sur le tag qui doit être lue ; max. 8 octets)	u8	rw	(→ 8.9.2)	-	oui
0x220A	0x01-0x40	Données du tag (données du tag 8 octets sont actualisées quand un nouveau tag entre dans la plage de lecture)	u64	ro		oui	-

Index	Sub-index	Nom (objet)	Type	Accès	Valeur par défaut	Capacité mapping PDO	Sauvegarder valeur de l'objet
Lecture de la zone de données							
0x2280	0x00	Début lecture de l'adresse (début de la zone d'adresse sur le tag qui doit être lue)	u16	rw	0x0000	-	oui
0x2281	0x00	Lecture longueur (longueur de la zone de mémoire sur le tag qui doit être lue)	u16	rw	0x0000	-	oui
0x2282	0x00	Données du tag (données du tag souhaitées configurées comme dans les objets 0x2280 et 0x2281)	dom	ro		-	-
Ecriture des données mappables							
0x2300	0x01-0x40	Ecriture de l'adresse de démarrage (début de la zone d'adresse sur le tag qui doit être écrite)	u16	rw	(→ 8.9.1)	-	oui
0x2301	0x01-0x40	Ecriture de la longueur (longueur de la zone de mémoire sur le tag qui doit être écrite ; max. 8 octets)	u8	rw	(→ 8.9.1)	-	oui
0x2302	0x01-0x40	Ecriture auto (activer l'accès en écriture automatique quand nouveau tag est détecté)	bool	rw	0	-	oui
0x230A	0x01-0x40	Données du tag (données du tag 8 octets)	u64	rww		oui	-
0x230F	0x00	Trigger écriture	u64	rww	0x00000000 00000000	oui	

Index	Sub-index	Nom (objet)	Type	Accès	Valeur par défaut	Capacité mapping PDO	Sauvegarder valeur de l'objet
Ecriture de la zone de données							
0x2380	0x00	Ecriture de l'adresse de démarrage (début de la zone d'adresses sur le tag qui doit être écrite)	u16	rw	0x0000	-	oui
0x2381	0x00	Ecriture de la longueur (longueur de la zone de mémoire sur le tag qui doit être écrite)	u16	rw	0x0000	-	oui
0x2382	0x00	Données du tag (données à écrire sur le tag configurées comme dans les objets 0x2380 et 0x2381)	dom	wo		-	-

Index	Sub-index	Nom (objet)	Type	Accès	Valeur par défaut	Capacité mapping PDO	Sauvegarder valeur de l'objet
Verrouillage de la zone de données							
0x2480	0x00	Verrouiller l'adresse de démarrage (début de la zone d'adresse sur le tag qui doit être verrouillée ; doit être alignée sur les zones tag)	u16	rw	0x0000	-	oui
0x2481	0x00	Verrouillage de la longueur (longueur de la zone de mémoire sur le tag à verrouiller ; doit être alignée sur les zones tag)	u16	rw	0x0000	-	oui
0x2482	0x00	Trigger verrouillage (trigger pour verrouiller des données sur le tag comme configuré dans les objets 0x2480 et 0x2481)	bool	wo		-	-
Filtre UID							
0x4603	0x00	Profondeur filtre UID	s8	rw	0x00	-	oui
0x4605	0x00	Profondeur filtre ID zéro	s8	rw	0x02	-	oui

8.6 Messages d'erreur

L'appareil supporte une série de messages de sécurité, transmis en cas d'un défaut de communication, de matériel ou RFID. Si une de ces erreurs se produit, le registre d'erreurs (index OV 0x1001) et le champ d'erreurs prédéfini (index OV 0x1003) sont mis à jour.

Le COB ID des messages de sécurité peut être modifié dans l'objet "COB-ID EMCY" (index RO 0x1014). Par l'activation du bit 31 dans cet objet, les messages de sécurité sont désactivés.

La durée de désactivation entre deux messages de sécurité peut être réglée via l'objet 0x1015. L'indication se fait en étapes de 100 µs.



Le COB ID des messages de sécurité est pré-réglé à 0x80 + Node ID.

Code d'erreur urgence	Registre d'erreurs (0x1001)	Code d'erreur fabricant	Nom d'erreur fabricant	Description cas d'urgence
0x8210	0x11			Protocole - PDO n'est pas traité dû aux fautes de longueur
0x8130	0x01			Surveillance - fautes Node Guarding ou Heartbeat
0x8100	0x11			Surveillance - erreurs de communication générales, à transmettre en cas de "bus off"
0x5000	0x81	0x01		Erreur matériel appareil (erreur d'antenne)
0x4200	0x09	0x02		Température de l'appareil trop élevée
0xFF00	0x81	0x01	RX: ISO_COMMAND_ERROR_NO_RESPONSE	Tag n'a pas répondu, peut-être que le tag n'est plus dans le champ ?
0xFF00	0x81	0x02	RX: ISO_COMMAND_ERROR_RX_ERROR	Erreur lors de la réception de la réponse du tag (erreur CRC, erreur framing, collision, etc.)
0xFF01	0x81	0x01	TX: ISO_COMMAND_ERROR_NO_RESPONSE	Tag n'a pas répondu, peut-être que le tag n'est plus dans le champ ?
0xFF01	0x81	0x02	TX: ISO_COMMAND_ERROR_RX_ERROR	Erreur lors de la réception de la réponse du tag (erreur CRC, erreur framing, collision, etc.)
0xFF02	0x81	0x01	ISO_TAG_ERROR_COMMAND_NOT_SPECIFIED	La commande indiquée n'est pas supportée. Exemple : commande erreur de code

Code d'erreur urgence	Registre d'erreurs (0x1001)	Code d'erreur fabricant	Nom d'erreur fabricant	Description cas d'urgence
0xFF02	0x81	0x02	ISO_TAG_ERROR_COMMAND_SYNTAX	Erreur de syntaxe de la commande. Nombre de zones trop élevé. Exemple : erreur de format
0xFF02	0x81	0x03	ISO_TAG_ERROR_OPTION_NOT_SUPPORTED	Les options indiquées ne sont pas supportées
0xFF02	0x81	0x0F	ISO_TAG_ERROR_OTHER	D'autres erreurs
0xFF02	0x81	0x10	ISO_TAG_ERROR_BLOCK_NOT_USABLE	La zone indiquée ne peut pas être utilisée (ou n'était pas trouvée)
0xFF02	0x81	0x11	ISO_TAG_ERROR_BLOCK_ALREADY_BLOCKED	La zone indiquée est bloquée et ne peut pas être bloquée une nouvelle fois
0xFF02	0x81	0x12	ISO_TAG_ERROR_BLOCK_NOT_UPDATABLE	La zone indiquée est bloquée et son contenu ne peut pas être actualisé
0xFF02	0x81	0x13	ISO_TAG_ERROR_BLOCK_WRITE_VERIFY	La zone indiquée ne pouvait pas être programmée normalement (erreur vérification d'écriture)
0xFF02	0x81	0x14	ISO_TAG_ERROR_BLOCK_LOCK_VERIFY	La zone indiquée ne pouvait pas être bloquée normalement (erreur vérification de verrouillage)
0xFF03	0x81	0x00	STATUS_BUFFER_OVERFLOW	Débordement du tampon interne

FR

8.7 Surveillance d'activité avec Heartbeat

Grâce à la fonctionnalité Heartbeat, l'activité d'un appareil dans le réseau CANopen peut être surveillée par le maître. L'appareil transmet, à intervalles réguliers, un message Heartbeat qui contient l'état de l'appareil.

La fonction Heartbeat est activée par la saisie d'une valeur supérieure à "0" dans l'objet intervalle de temps du Heartbeat (index RO 0x1017). Cette valeur spécifie le temps entre deux signaux Heartbeat en millisecondes. La valeur "0" désactive la fonction Heartbeat.

8.8 Modification d'objets

Des modifications aux objets dans le répertoire d'objets sont validées immédiatement. Lors d'un reset toutes les modifications sont perdues. Pour éviter cela, les objets doivent être sauvegardés dans la mémoire rémanente interne (flash). Tous les objets marqués "Sauvegarder valeur de l'objet : oui" dans le répertoire d'objets sont mémorisés en permanence dans le flash de l'appareil. Grâce à l'écriture de la signature "save" (65766173h) pour sauvegarder les objets (index RO 1010h/01h), tous les objets actuels du répertoire d'objets sont transmis dans la mémoire flash.

Les objets peuvent être remis aux réglages usine par l'écriture de la signature "load" (64616F6Ch) dans l'index RO 1011h/01h. Les modifications seront validées après un reset.

Selon l'architecture du réseau CANopen, les objets peuvent également être mémorisés de façon centralisée dans un maître CANopen. Dans ce cas, les objets seront transmis à l'appareil lors du démarrage du système et les valeurs sauvegardées localement seront remplacées.

 Particularités des objets Node ID (index RO 0x20F0 et 0x20F1) et Bit-rate (index RO 0x20F2 et 0x20F3) :

- Une modification de ces objets ne sera validée qu'après un reset (→ 8.2 Modification du Node ID et du Bit-rate).
- Les objets ne peuvent pas être transférés en flash via l'index RO 1010h/01h.
- Les objets ne peuvent pas être remis aux réglages usine via l'index RO 1011h/01h.

8.9 Objets de données process

64 objets de données process émetteurs et récepteurs sont disponibles. A la livraison, 4 objets de données process sont actifs.

8.9.1 Objets de données process émetteurs (TPDO)

Le tableau suivant contient les objets de données process émetteurs (TPDO) à la livraison.

TPDO	Réglages pour mapping PDO	Répertoire objets			Mémoire tag	
	COB	Index objet mappé	Sub-index objet mappé	Longueur d'objet mappée	Lecture adresse de départ	Lecture longueur
1	Node ID + 0x0180	0x2150	0x00	0x20	Etat d'appareil	
2	Node ID + 0x0280	0x220A	0x01	0x40	0x00000000	0x08
3	Node ID + 0x0380	0x220A	0x02	0x40	0x00000008	0x08
4	Node ID + 0x0480	0x220A	0x03	0x40	0x00000010	0x08
5	0 (désactivé)	0x220A	0x04	0x40	0x00000018	0x08
64	0 (désactivé)	0x220A	0x3F	0x04	0x000001F0	0x08

8.9.2 Objets de données process récepteurs (RPDO)

Le tableau suivant contient les objets de données process récepteurs (RPDO) à la livraison.

RPDO	Réglages pour mapping PDO	Répertoire objets			Mémoire tag	
	COB	Index objet mappé	Sub-index objet mappé	Longueur d'objet mappée	Ecriture de l'adresse de départ	Ecriture de la longueur
1	Node ID + 0x0200	0x230F	0x00	0x40	Ecriture trigger	
2	Node ID + 0x0300	0x230A	0x01	0x40	0x00000000	0x08
3	Node ID + 0x0400	0x230A	0x02	0x40	0x00000008	0x08
4	Node ID + 0x0500	0x230A	0x03	0x40	0x00000010	0x08
5	0 (désactivé)	0x230A	0x04	0x40	0x00000018	0x08
64	0 (désactivé)	0x230A	0x3F	0x04	0x000001F8	0x08

8.10 Etat d'appareil

L'état actuel de l'appareil est indiqué dans l'objet "Etat d'appareil" (index RO 0x2150, sub-index 0x00). A la livraison, l'objet TPDO1 est attribué.

Bit	31	30	29	28	27	26	25	24
Etat	tag_err							
Valeur par défaut	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Etat	write_err							
Valeur par défaut	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Etat	read_err							
Valeur par défaut	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Etat	r	r	buf_ovfl	fr_err	busy	present	ant	pow
Valeur par défaut	0	0	0	0	0	0	1	1

Etat	Valeur	Description	Message EMCY
pow	1	Opérationnel (toujours 1)	
ant	0	Antenne désactivée	
	1	Antenne activée	
present	0	Aucun tag disponible	
	1	Tag disponible	
busy	0	Etat de repos	
	1	Accès en lecture ou écriture actif	
fr_err	0	Frontend OK	
	1	Erreur frontend détectée (problème matériel)	oui
buf_ovfl	0	Tampon OK	
	1	Débordement du tampon détecté	oui
read_err		Erreur de la dernière opération de lecture	oui
write_err		Erreur de la dernière opération d'écriture	oui

Etat	Valeur	Description	Message EMCY
tag_err		Message d'erreur tag pour la dernière opération	oui

Lecture des codes d'erreur (actualisés après chaque accès en lecture du tag)		
0x00	ISO_COMMAND_ERROR_NO_ERROR	Aucune erreur, commande réussie
0x01	ISO_COMMAND_ERROR_NO_RESPONSE	Tag n'a pas répondu, peut-être que le tag n'est plus dans le champ
0x02	ISO_COMMAND_ERROR_RX_ERROR	Erreur lors de la réception de la réponse du tag (erreur CRC, erreur framing, collision, etc.)

Ecriture des codes d'erreurs (actualisés après chaque accès en écriture du tag)		
0x00	ISO_COMMAND_ERROR_NO_ERROR	Aucune erreur, commande réussie
0x01	ISO_COMMAND_ERROR_NO_RESPONSE	Tag n'a pas répondu, peut-être que le tag n'est plus dans le champ ?
0x02	ISO_COMMAND_ERROR_RX_ERROR	Erreur lors de la réception de la réponse du tag (erreur CRC, erreur framing, collision, etc.)

Codes d'erreur tag (actualisés après accès en lecture / écriture du tag)		
0x00	ISO_TAG_ERROR_NO_ERROR	Aucune erreur du tag
0x01	ISO_TAG_ERROR_COMMAND_NOT_SPECIFIED	La commande indiquée n'est pas supportée. Exemple : commande erreur de code
0x02	ISO_TAG_ERROR_COMMAND_SYNTAX	Erreur de syntaxe de la commande. Nombre de zones trop élevé. Exemple : erreur de format
0x03	ISO_TAG_ERROR_OPTION_NOT_SUPPORTED	Les options indiquées ne sont pas supportées
0x0F	ISO_TAG_ERROR_OTHER	D'autres erreurs
0x10	ISO_TAG_ERROR_BLOCK_NOT_USABLE	La zone indiquée ne peut pas être utilisée (ou n'était pas trouvée)
0x11	ISO_TAG_ERROR_BLOCK_ALREADY_BLOCKED	La zone indiquée est bloquée et ne peut pas être bloquée une nouvelle fois
0x12	ISO_TAG_ERROR_BLOCK_NOT_UPDATEABLE	La zone indiquée est bloquée et son contenu ne peut pas être actualisé
0x13	ISO_TAG_ERROR_BLOCK_WRITE_VERIFY	La zone indiquée ne pouvait pas être programmée normalement (erreur vérification d'écriture)
0x14	ISO_TAG_ERROR_BLOCK_LOCK_VERIFY	La zone indiquée ne pouvait pas être bloquée normalement (erreur vérification de verrouillage)

8.11 Désactiver l'antenne

L'antenne de l'appareil peut être désactivée si l'objet "antenne active" (index RO 0x2151) est réglé à la valeur 0. Dans ce cas, plus aucun tag n'est détecté, puisque le champ magnétique de l'appareil n'est plus actif.

L'antenne est réactivée avec la valeur 1. Avec l'objet "antenne active", des interférences entre des appareils positionnés l'un à côté de l'autre peuvent être évitées en désactivant les antennes des deux appareils en alternance.

8.12 Sélectionner le type de tag

L'appareil est compatible avec plusieurs types de tag selon ISO15693. Selon la taille de la mémoire de données utilisables et le fabricant, les tag diffèrent en ce qui concerne l'accès aux données. C'est pourquoi l'appareil doit connaître le type de tag utilisé dans le système.

Dans l'objet 0x2161, le type de tag utilisé dans le système RFID peut être sélectionné. Les types de tag disponibles peuvent être lus dans l'objet 0x2180, sub-index 0x01-0xFE.

Type de tag	Nom	Taille des blocs [octets]	Nombre de blocs
1	défini par l'utilisateur	?	?
2	I-Code SLI	4	28
3	I-Code SLI-S	4	40
4	I-Code SLI-L	4	8
5	F-MEM 2k	8	250
6	F-MEM 232b	4	58
7	F-MEM 8k	32	256
8	TI_32b	4	8
9	TI_256b	4	64
10	ST_128b	4	32
11	ST_256b	4	64
12	ST_8k	4	2048
13	I-Code SLIX2	4	79

Via l'objet 0x2182 0x06, le type de tag détecté par l'appareil peut être vérifié. Pour ce faire, le type de tag détecté doit être lu dans l'objet 0x2182 sub-index 0x06 et cette valeur doit ensuite être saisie dans l'objet 0x2161.

Le type de tag 1 est d'importance particulière : Les paramètres "Taille des blocs" et "Nombre de blocs" sont déterminés par l'appareil même. Si les paramètres ne conviennent pas aux types de tag connus, le type 1 "défini par l'utilisateur" est utilisé.



La détection du type de tag n'est pas supportée par tous les tag.

 Le type de tag réglé n'est sauvegardé en permanence dans l'appareil que si l'objet "sauvegarder paramètre" est utilisé (→ 8.8 Modification d'objets).

 Le type de tag 2 est pré-réglé.

8.13 Lecture des informations d'un tag

Via les objets 0x2180 à 0x2182, les informations d'un tag peuvent être lues. Pour ce faire, le tag doit se trouver dans la zone de détection de l'appareil.

Les objets 0x2180 et 0x2182 ne sont valables que tant que le tag est détecté. Si aucun tag n'est à portée, les valeurs des objets seront remises à 0.

La valeur de l'objet 0x2182 peut être lue par le tag sur demande.

 La lecture d'informations n'est pas supportée par tous les types de tag.

8.14 Valeur RSSI

La valeur RSSI (Received Signal Strength, index RO 0x2162) indique l'intensité du signal reçu qui est émis par le tag devant l'appareil :

0 : aucun tag détecté

1 : l'intensité de réception min.

8 : l'intensité de réception max.

 L'intensité de réception max. n'est atteinte que par quelques combinaisons appareil / tag.

 L'intensité de réception dépend de la distance entre le tag et la face active de l'appareil.

 Des changements de positions dans l'environnement, par exemple d'objets métalliques, peuvent influencer l'intensité de réception.

8.15 Filtre de détection tag

Les situations suivantes causent une détection et lecture multiple du tag non souhaitées :

- Le tag se trouve dans les limites de la portée.
- Les conditions de montage ont un effet défavorable sur le champ électromagnétique de l'appareil.

Par conséquent le tag n'est pas détecté sans équivoque ce qui entraîne des messages d'erreur lors de la lecture et de l'écriture par PDO. Avec les objets "Profondeur filtre UID" et "Profondeur filtre ID zéro" les messages d'erreur peuvent être filtrés.



Les valeurs suivantes ont fait leurs preuves :

- "0" à "5" pour les applications dynamiques (tag au passage vite)
- ">5" pour les applications statiques

Temps [ms]	0	7	14	21	28	35	42	49	56	63	70	77	84	91	98	105	112	119	126	133	140	
Tag dans le champ		■	■	■				■	■	■	■	■	■	■	■							
Tag n'est pas dans le champ	■				■	■	■								■	■	■	■	■	■	■	■
Profondeur filtre UID: 0, profondeur filtre ID zéro : 0																						
Tag détecté		■	■	■				■	■	■	■	■	■	■	■							
Tag ne pas détecté	■				■	■	■								■	■	■	■	■	■	■	■
Profondeur filtre UID: profondeur filtre ID zéro: 0																						
Tag détecté														■	■							
Tag ne pas détecté	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Profondeur filtre UID: 0, profondeur filtre ID zéro : 5																						
Tag détecté		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Tag ne pas détecté	■																				■	■
Profondeur filtre UID: profondeur filtre ID zéro: 5																						
Tag détecté														■	■	■	■	■	■	■	■	■
Tag ne pas détecté	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

8.15.1 Objet profondeur filtre UID

Avec l'objet "profondeur filtre UID" (0x4603/0x00) le nombre de bonnes détections tag que l'appareil doit exécuter est réglé. Quand le nombre réglé est atteint, le tag sur le bus CAN est considéré comme détecté (tag présent).

La valeur "0" désactive le filtre. Les valeurs ">0" retardent le bit "Tag présent" de 7 ms. Ainsi une temporisation à l'enclenchement de la valeur tag est réalisée. La détection dans les limites de la portée se stabilise parce qu'une valeur n'est fournie que si le tag est détecté de façon stable.

8.15.2 Objet profondeur filtre ID zéro

Avec l'objet "Profondeur filtre ID zéro" (0x4605/0x00) le nombre des détections tag non réussies que l'appareil doit exécuter se termine. Quand le nombre réglé est atteint, le tag sur le bus CAN est considéré comme ne plus présent (tag présent).

La valeur "0" désactive le filtre. Les valeurs ">0" retardent la remise du bit "Tag présent" de 7 ms. Ainsi une temporisation au déclenchement de la valeur tag est réalisée. La détection dans les limites de la portée se stabilise parce qu'une valeur n'est fournie que si le tag reste non-détecté de façon stable.

9 Transmission de données avec le tag

9.1 Lecture de l'UID (Unique Identification Number) du tag

L'UID du tag est disponible dans l'objet 0x2180 dès qu'un tag se trouve dans la portée de lecture de l'appareil. Si aucun tag n'est disponible, la valeur 0x0000000000000000 est retournée.

Si l'objet est mappé sur un TPDO, la transmission se fait de façon événementielle dès qu'un tag entre dans la portée de lecture ou est enlevé du champ de lecture.

9.2 Lecture de données du tag via transfert PDO

La transmission de données PDO du tag se fait de façon événementielle. Cela signifie que les TPDO configurés seront transmis automatiquement par l'appareil quand les données sont changées. Cela est par exemple le cas lorsqu'un nouveau tag entre dans la zone de détection de l'appareil. Les données sont lues indépendamment par le tag et transmises via le bus CAN à l'aide des TPDO.

Les données lues par le tag, qui peuvent être attribuées à un TPDO, se trouvent dans l'objet 0x220A, sub-index 0x01-0x40.



Seulement les données du tag qui sont attribuées à un TPDO sont lues. Des objets de données non attribués ne sont pas mis à jour automatiquement.

Pour chaque objet de données il y a deux objets qui servent pour la configuration : 0x2200 (lecture adresse de départ) et 0x2201 (lecture longueur) avec les sub-index appropriés pour l'objet de données. Dans les objets, l'adresse de départ dans la zone de données utiles du tag et la longueur des données à lire sont réglées.



Dans un TPDO, toujours 64 bits de données (8 octets) sont transmis. Si la longueur de données configurées est inférieure à 64 bits, les bits restants sont remplis avec 0.



Un maximum de 64 bits peut être transmis dans un TPDO. Si un plus grand nombre de données doit être transféré, d'autant plus de TPDO doivent être attribués et les objets de données correspondants doivent être configurés.

FR

9.2.1 Exemple 1

La zone de données 0x10 à 0x18 (8 octets) doit être transmise avec le deuxième TPDO.

	Réglages pour mapping PDO	Répertoire objets		
TPDO	COB	Index objet	Sub-index objet	Longueur objet
2	Node ID + 0x0280	0x220A	0x01	0x40

Répertoire objets			
Index	Sub-index	Nom (objet)	Valeur
0x2200	0x01	Début lecture de l'adresse (début de la zone d'adresse sur le tag qui doit être lue)	0x10
0x2201	0x01	Lecture longueur (longueur de la zone de mémoire sur le tag qui doit être lue ; max. 8 octets)	0x08

9.2.2 Exemple 2

La zone de données 0x44 à 0x48 (4 octets) doit être transmise avec le sixième TPDO.

	Réglages pour mapping PDO	Répertoire objets		
TPDO	COB	Index objet	Sub-index objet	Longueur objet
6	Node ID + 0x0680	0x220A	0x05	0x40

Répertoire objets			
Index	Sub-index	Nom (objet)	Valeur
0x2200	0x05	Début lecture de l'adresse (début de la zone d'adresse sur le tag qui doit être lue)	0x44

Répertoire objets			
Index	Sub-index	Nom (objet)	Valeur
0x2201	0x05	Lecture longueur (longueur de la zone de mémoire sur le tag qui doit être lue ; max. 8 octets)	0x04

9.3 Ecriture de données sur le tag via transfert PDO

Afin d'écrire des données sur un tag via transfert PDO, un RPDO doit être attribué à l'objet 0x230A avec un sub-index dans la zone de 0x01 à 0x40. L'adresse de la zone de données utiles du tag sur laquelle les données sont à écrire est déterminée dans l'objet 0x2300. Les sub-index de ces objets doivent être compatibles.

L'écriture sur le tag se fait après l'écriture des données sur le RPDO et la modification du bit correspondant dans l'objet "Ecriture trigger" (index RO 0x230F, sub-index 0x00).

	MSB								LSB
Bit	63	62	61	2	1	0
Trigger	tr64	tr63	tr62	tr3	tr2	tr1
Valeur par défaut	0	0	0	0	0	0	0	0	0

Trigger	Description
tr64	Trigger pour données tag 64 (0x230A/0x40)
tr63	Trigger pour données tag 63 (0x230A/0x3F)
tr62	Trigger pour données tag 62 (0x230A/0x3E)
tr61	Trigger pour données tag 61 (0x230A/0x3D)
tr60	Trigger pour données tag 60 (0x230A/0x3C)
tr59	Trigger pour données tag 59 (0x230A/0x3B)
tr58	Trigger pour données tag 58 (0x230A/0x3A)
...	...
tr6	Trigger pour données tag 6 (0x230A/0x6)
tr5	Trigger pour données tag 5 (0x230A/0x5)
tr4	Trigger pour données tag 4 (0x230A/0x4)
tr3	Trigger pour données tag 3 (0x230A/0x3)
tr2	Trigger pour données tag 2 (0x230A/0x2)
tr1	Trigger pour données tag 1 (0x230A/0x1)

L'opération d'écriture se fait toujours lors du changement de bit du bit correspondant (0->1 ou 1->0). Idéalement, l'objet "Ecriture trigger" (index RO 0x230F, sub-index 0x00) est attribué à un RPDO. A la livraison, l'objet "Ecriture trigger" est attribué au premier RPDO.

Avec l'objet "Ecriture auto" (index RO 0x2302), l'écriture automatique de données peut être activée. Dès qu'un tag se trouve dans la zone de détection, les dernières données seront écrites sur le tag.



Seulement des données jusqu'à la longueur de données configurée seront écrites sur le tag. Les données suivantes seront ignorées. Si plus de 8 octets sont à transmettre, plus de RPDO doivent être attribués et les objets de données correspondants doivent être configurés.

9.3.1 Exemple 1

La zone de données 0x10 à 0x18 (8 octets) doit être transmise avec le deuxième RPDO.

	Réglages pour mapping PDO	Répertoire objets		
RPDO	COB	Index objet	Sub-index objet	Longueur objet
2	Node ID + 0x0200	0x230A	0x01	0x40

Répertoire objets			
Index	Sub-index	Nom (objet)	Valeur
0x2300	0x01	Début lecture de l'adresse (début de la zone d'adresse sur le tag qui doit être lue)	0x10
0x2301	0x01	Lecture longueur (longueur de la zone de mémoire sur le tag qui doit être lue ; max. 8 octets)	0x08
0x2302	0x01	Ecriture auto	0x00

Transmission de données via RPDO :

Transmission PDO	PDO	Données
A l'appareil	RPDO	0x12345678

Démarrage de l'accès en écriture :

Transmission PDO	PDO	Données
A l'appareil	RPDO 1	Commutation de bit 0

9.3.2 Exemple 2

La zone de données 0x44 à 0x48 (4 octets) doit être transmise avec le sixième RPDO. De plus, les données doivent être écrites sur un tag chaque fois que le tag entre dans la zone de détection de l'appareil.

	Réglages mapping PDO	Répertoire objets		
RPDO	COB	Index objet	Sub-index objet	Longueur objet
6	Node ID + 0x0600	0x230A	0x05	0x40

Répertoire objets			
Index	Sub-index	Nom (objet)	Valeur
0x2300	0x05	Début lecture de l'adresse (début de la zone d'adresse sur le tag qui doit être lue)	0x44
0x2301	0x05	Lecture longueur (longueur de la zone de mémoire sur le tag qui doit être lue ; max. 8 octets)	0x04
0x2302	0x05	Ecriture auto	0x01

Transmission de données via RPDO :

Transmission PDO	PDO	Données
A l'appareil	RPDO 6	0x12340000

Les données sont écrites sur le tag lorsque ce dernier entre dans la zone de détection.



64 bits de données (8 octets) doivent toujours être transmis à un RPDO. Si la longueur de données configurée est inférieure à 64 bits, les bits restants seront ignorés.

9.4 Traitement d'erreurs en cas de transfert PDO

Si un accès en lecture / écriture sur un tag n'est pas possible, l'appareil génère un message de sécurité sur le bus CAN.

Le code d'erreur peut être lu du registre d'erreurs (index RO 0x1001, sub-index 0x00) et du champ d'erreurs prédéfini (index RO 0x1003, sub-index 0x01-0x02) (→ 8.6 Messages d'erreur).

9.5 Lecture des données du tag via transfert SDO

Afin de lire les données d'un tag via un transfert SDO, l'adresse et la longueur des données sur le tag doivent être définies. L'adresse doit être indiquée dans l'objet 0x2280 et la longueur des données dans l'objet 0x2281.

Ensuite, l'accès en lecture du tag peut être démarré par un transfert de données vers l'objet 0x2282.

9.5.1 Exemple

La zone de données 0x50 à 0x70 doit être lue par le tag.

Répertoire objets			
Index	Sub-index	Nom (objet)	Valeur
0x2280	0x00	Début lecture de l'adresse (début de la zone d'adresse sur le tag qui doit être lue)	0x50
0x2281	0x00	Lecture longueur (longueur de la zone de mémoire sur le tag qui doit être lue ; max. 8 octets)	0x20

Le transfert est démarré par la lecture de l'objet 0x2282, sub-index 0x00.

 Les données sont transmises en une seule pièce en tant que type de données Domain. Jusqu'à une longueur de données de 4 octets, le transfert se fait en tant que "expedited", au-dessus en tant que "segmented".

 Le récepteur doit être prêt pour la mémorisation temporaire et le traitement des données.

9.6 Ecriture de données sur le tag via transfert SDO

Afin d'écrire les données sur un tag via un transfert SDO, l'adresse et la longueur des données sur le tag doivent être définies.

L'adresse doit être indiquée dans l'objet 0x2380 et la longueur des données dans l'objet 0x2381. Ensuite, l'accès en écriture sur le tag peut être démarré par un transfert de données vers l'objet 0x2382.

9.6.1 Exemple

La zone de données 0x34 à 0x37 doit être transférée sur le tag.

Répertoire objets			
Index	Sub-index	Nom (objet)	Valeur
0x2380	0x00	Début écriture adresse (début de la zone d'adresse sur le tag qui doit être écrite)	0x34
0x2381	0x00	Ecriture de la longueur (longueur de la zone de mémoire sur le tag qui doit être écrite)	0x03
0x2382	0x00	Données du tag (données qui doivent être écrites sur le tag)	0x010203

 Les données sont transmises en une seule pièce en tant que type de données Domain. Jusqu'à une longueur de données de 4 octets, le transfert se fait en tant que "expedited", au-dessus en tant que "segmented".



L'émetteur doit pouvoir fournir la quantité de données indiquée.

9.7 Blocage des zones de données sur le tag via transfert SDO

Les zones de données du tag peuvent être équipées d'une protection en écriture.



La protection en écriture d'une zone de données ne peut pas être enlevée.

Dans l'objet "Verrouiller le début de l'adresse" (index RO 0x2480), l'adresse de départ de la zone de données à protéger est mémorisée. En plus, la longueur de la zone de données est mémorisée dans l'objet "Verrouillage de la longueur" (index RO 0x2481).



L'adresse de départ doit correspondre à l'adresse de départ d'un bloc de mémoire sur le tag. La longueur doit être un multiple de la longueur d'un bloc de mémoire sur le tag.

Pour activer la protection en écriture, le trigger (index RO 0x2482) est réglé à 1.

9.7.1 Exemple

La zone de données 0x04 à 0x0C doit être protégée en écriture dans un tag avec la taille des blocs 4 (2 blocs ou 8 octets).

Répertoire objets			
Index	Sub-index	Nom (objet)	Valeur
0x2480	0x00	Verrouiller le début de l'adresse (début de la zone d'adresse sur le tag qui doit être verrouillée)	0x04
0x2481	0x00	Ecriture de la longueur (longueur de la zone de mémoire sur le tag qui doit être verrouillée)	0x08
0x2482	0x00	Tag trigger verrouillage	0x01

9.8 Traitement d'erreurs en cas de transfert SDO

Les transferts SDO sont des transferts validés. Lorsqu'un défaut se produit pendant le transfert ou les actions déclenchées par le transfert, un défaut est signalé après le transfert SDO.

Code d'erreur SDO	Description	Cause possible
0x05030000	Bit toggle inchangé.	
0x05040000	Protocole SDO expiré.	
0x05040001	Spécification commande client/serveur non valable ou inconnue.	
0x05040002	Largeur de la zone non valable (uniquement mode bloc).	
0x05040003	Numéro séquence non valable (uniquement mode bloc).	
0x05040004	Erreur CRC (uniquement mode bloc).	
0x05040005	Aucun emplacement de mémoire libre.	
0x06010000	Accès non supporté sur un objet.	
0x06010001	Essai de lire un objet seulement écriture.	
0x06010002	Essai d'écrire un objet seulement lecture.	
0x06020000	Objet n'existe pas dans le dictionnaire objet.	
0x06040041	Objet ne peut pas être mappé sur PDO.	
0x06040042	Le nombre et la longueur des objets mappés devraient dépasser la longueur PDO.	
0x06040043	Raison incompatibilité générale paramètres.	
0x06040047	Incompatibilité générale paramètres dans l'appareil.	
0x06060000	Accès non réussi à cause d'un défaut du matériel.	
0x06070010	Type de données n'est pas identique, longueur du paramètre service n'est pas identique.	
0x06070012	Type de données n'est pas identique ; paramètre de service trop long.	
0x06070013	Type de données n'est pas identique ; paramètre de service trop court.	
0x06090011	Sub-index n'existe pas.	
0x06090030	Valeur non valide pour paramètre (uniquement téléchargement).	
0x06090031	Valeur du paramètre écrit trop haute (uniquement téléchargement).	
0x06090032	Valeur du paramètre écrit trop basse (uniquement téléchargement).	
0x06090036	Valeur maximale est plus basse que la valeur minimum.	
0x060A0023	Ressource non disponible : connexion SDO.	
0x08000000	Erreur générale.	

Code d'erreur SDO	Description	Cause possible
0x08000020	Les données ne peuvent pas être transmises à l'application ou sauvegardées.	Erreur accès en lecture ou écriture du tag. Des informations détaillées dans l'objet état appareil (0x2150).
0x08000021	Les données ne peuvent pas être transmises à l'application ou sauvegardées à cause d'une commande locale.	
0x08000022	Les données ne peuvent pas être transmises à l'application ou sauvegardées à cause de l'état actuel de l'appareil.	
0x08000023	Génération dynamique du répertoire objets non réussie ou répertoire objets ne pas présent (par ex. répertoire objets est généré du fichier et la génération échoue à cause d'une erreur fichier).	
0x08000024	Aucune donnée disponible.	Longueur de données = 0

10 Données EDS

Le fichier EDS sert de modèle pour les différentes configurations d'un type d'appareil. Un fichier DCF est généré du fichier EDS, qui contient la configuration de l'appareil, les valeurs de l'objet, le Node ID et le Bit-rate.

Pour la configuration du réseau CANopen et des appareils, des outils de configuration CANopen sont disponibles.

Le fichier EDS peut être téléchargé du site web ifm :

www.ifm.com

Contenu du fichier EDS :

- Fonctions de communication et objets (selon le profil CANopen DS-301)
- Objets spécifiques au fabricant



L'installation du fichier EDS dépend de votre outil de configuration. Si nécessaire, veuillez contacter le fabricant de votre commande.

11 Maintenance, réparation et élimination

- ▶ Ne pas ouvrir l'appareil. Aucune opération de maintenance ne peut être effectuée par l'utilisateur. L'appareil ne doit être réparé que par le fabricant.
- ▶ Respecter la réglementation du pays en vigueur pour la destruction écologique de l'appareil.

12 Glossaire

Terme	Description
0b ...	Valeur numérique binaire (pour le codage des bits), par ex. 0b0001 0000
0x ...	Valeur numérique hexadécimale, par ex. 0x64 (= 100 décimale)
AFI	Indication du champ d'application du tag
CAN	Controller Area Network (système bus pour les engins mobiles)
CAN_H	CAN haut ; raccord/câble CAN avec un haut niveau de tension
CAN_L	CAN bas ; raccord/câble CAN avec un bas niveau de tension
CANopen	Protocole réseau basé sur CAN sur le niveau application avec une interface de communication ouverte (répertoire objets)
CiA	CAN in Automation e.V. (organisation regroupant des fabricants et utilisateurs en Allemagne / Erlangen, organe de définition et contrôle pour CAN et des protocoles réseaux basés sur CAN)
COB	Objet de communication CANopen : PDO, SDO, EMCY, ...
COB ID	Numéro d'identification objet de communication pour l'affectation des paquets de données dans le réseau CANopen
DSFID	Numéro d'identification pour l'affectation de la structure de données sur un tag
EDS	Fiche technique électronique
EMCY Object	Objet d'urgence (message d'alarme ; appareil signale une erreur)
Messages de sécurité	Messages sur le bus CANopen pour la signalisation de défauts
Error Reg	Registre d'erreurs (saisie avec une reconnaissance d'erreur)
Heartbeat	Surveillance cyclique paramétrable parmi les participants réseau. Contrairement au "Node Guarding" le maître NMT supérieur n'est pas nécessaire.
ID	Identifiant qui marque un message CAN. La valeur numérique de l'ID contient en même temps une priorité concernant l'accès bus (ID 0 = niveau de priorité le plus haut).
Identifiant	Voir ID
Tag	Tag RFID
LSS	Procédure de réglage de paramètres fondamentaux d'appareils
NMT	Gestion du réseau
NODE ID	Numéro unique d'un participant dans le réseau CANopen
Objet / OBJ	Terme générique pour les données/messages interchangeables à l'intérieur du réseau CANopen
RO	Répertoire objets
PDO	Process Data Object ; dans le réseau CANopen, pour la transmission des données process en temps réel, comme par ex. la vitesse de rotation d'un moteur. Les PDO ont un niveau de priorité plus haut que les SDO ; contrairement aux SDO ils sont transférés sans confirmation. Les PDO consistent en un message CAN avec identifiant et des données utiles jusqu'à 8 octets.
Mapping PDO	Décrit les données applications qui sont transférées avec un PDO.
ro	Unidirectionnel ; seulement lecture

Terme	Description
RPDO	Objet de données process, reçu par l'appareil
RSSI	L'intensité de réception
rw	Bidirectionnel; lecture - écriture
SDO	Avec cet objet, un répertoire objets d'un participant réseau est atteint de façon ciblée (lire / écrire). Un SDO peut consister en plusieurs messages CAN. La transmission de messages individuels est confirmée par le participant adressé. Utilisant les SDO, les appareils peuvent être configurés et paramétrés.
SYNC	Le télégramme SYNC déclenche la transmission de données process.
TPDO	Objet de données process, transmis par l'appareil
UID	Numéro d'identification unique d'un tag
wo	Unidirectionnel, seulement écriture