

SIEMENS

SIMOTION

Complément pour moteur de
positionnement
SIMODRIVE POSMO A

Description fonctionnelle

Avant-propos

Description

1

Blocs fonctionnels

2

Exemple d'application

3

Annexe

A

Mentions légales

Signalétique d'avertissement

Ce manuel donne des consignes que vous devez respecter pour votre propre sécurité et pour éviter des dommages matériels. Les avertissements servant à votre sécurité personnelle sont accompagnés d'un triangle de danger, les avertissements concernant uniquement des dommages matériels sont dépourvus de ce triangle. Les avertissements sont représentés ci-après par ordre décroissant de niveau de risque.

 DANGER
signifie que la non-application des mesures de sécurité appropriées entraîne la mort ou des blessures graves.

 ATTENTION
signifie que la non-application des mesures de sécurité appropriées peut entraîner la mort ou des blessures graves.

 PRUDENCE
accompagné d'un triangle de danger, signifie que la non-application des mesures de sécurité appropriées peut entraîner des blessures légères.

PRUDENCE
non accompagné d'un triangle de danger, signifie que la non-application des mesures de sécurité appropriées peut entraîner un dommage matériel.

IMPORTANT
signifie que le non-respect de l'avertissement correspondant peut entraîner l'apparition d'un événement ou d'un état indésirable.

En présence de plusieurs niveaux de risque, c'est toujours l'avertissement correspondant au niveau le plus élevé qui est reproduit. Si un avertissement avec triangle de danger prévient des risques de dommages corporels, le même avertissement peut aussi contenir un avis de mise en garde contre des dommages matériels.

Personnes qualifiées

L'installation et l'exploitation de l'appareil/du système concerné ne sont autorisées qu'en liaison avec la présente documentation. La mise en service et l'exploitation d'un appareil/système ne doivent être effectuées que par des **personnes qualifiées**. Au sens des consignes de sécurité figurant dans cette documentation, les personnes qualifiées sont des personnes qui sont habilitées à mettre en service, à mettre à la terre et à identifier des appareils, systèmes et circuits en conformité avec les normes de sécurité.

Utilisation des produits Siemens conforme à leur destination

Tenez compte des points suivants:

 ATTENTION
Les produits Siemens ne doivent être utilisés que pour les cas d'application prévus dans le catalogue et dans la documentation technique correspondante. S'ils sont utilisés en liaison avec des produits et composants d'autres marques, ceux-ci doivent être recommandés ou agréés par Siemens. Le fonctionnement correct et sûr des produits suppose un transport, un entreposage, une mise en place, un montage, une mise en service, une utilisation et une maintenance dans les règles de l'art. Il faut respecter les conditions d'environnement admissibles ainsi que les indications dans les documentations afférentes.

Marques de fabrique

Toutes les désignations repérées par ® sont des marques déposées de Siemens AG. Les autres désignations dans ce document peuvent être des marques dont l'utilisation par des tiers à leurs propres fins peut enfreindre les droits de leurs propriétaires respectifs.

Exclusion de responsabilité

Nous avons vérifié la conformité du contenu du présent document avec le matériel et le logiciel qui y sont décrits. Ne pouvant toutefois exclure toute divergence, nous ne pouvons pas nous porter garants de la conformité intégrale. Si l'usage de ce manuel devait révéler des erreurs, nous en tiendrons compte et apporterons les corrections nécessaires dès la prochaine édition.

Avant-propos

Contenu du manuel des fonctions

Le présent **document** fait partie intégrante de la **documentation Programmation SIMOTION - Références**.

Cette documentation est un complément à la documentation SIMODRIVE POSMO A, Manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*.

Cette documentation est fournie avec SIMOTION SCOUT en tant que document électronique.

Le manuel présent décrit comment commander et paramétrer un entraînement POSMO A à partir d'un programme SIMOTION à l'aide de blocs fonctionnels.

Le manuel des fonctions décrit les différences de commande et de paramétrage d'un entraînement POSMO A à l'aide d'un système SIMOTION en comparaison avec un système SIMATIC.

Bloc fonctionnel

Les blocs fonctionnels pour la communication entre le système SIMOTION le moteur de positionnement décentralisé SIMODRIVE POSMO A sont des constituants de la bibliothèque de commandes du système d'ingénierie "SIMOTION SCOUT".

Documentation SIMOTION

Une vue d'ensemble de la documentation SIMOTION est donnée dans une bibliographie séparée.

Cette documentation est fournie avec SIMOTION SCOUT en tant que document électronique.

La documentation SIMOTION comporte 9 paquets de documentation qui contiennent environ 80 documents SIMOTION et des documents relatifs aux systèmes apparentés (SINAMICS p. ex.).

Les paquets de documentation suivants sont disponibles pour la version produit SIMOTION V4.1 SP3 :

- Manuel du système d'ingénierie SIMOTION
- SIMOTION Description du système et des fonctions
- SIMOTION Diagnostic
- SIMOTION Programmation
- SIMOTION Programmation - Références
- SIMOTION C
- SIMOTION P350
- SIMOTION D4xx
- SIMOTION Documentation complémentaire

Assistance téléphonique et adresses Internet

Support technique

Pour toutes vos questions techniques, adressez-vous au service d'assistance téléphonique :

	Europe / Afrique
Téléphone	+49 180 5050 222 (payant)
Fax	+49 180 5050 223
Internet	http://www.siemens.com/automation/support-request

	Amérique
Téléphone	+1 423 262 2522
Fax	+1 423 262 2200
E-mail	mailto:techsupport.sea@siemens.com

	Asie / Pacifique
Téléphone	+86 1064 719 990
Fax	+86 1064 747 474
E-mail	mailto:adsupport.asia@siemens.com

Remarque

Pour tout conseil technique, vous trouverez les coordonnées téléphoniques spécifiques à chaque pays sur Internet :

<http://www.siemens.com/automation/service&support>

Les appels sont payants, par exemple 0,14 €/min depuis le réseau fixe allemand. Les tarifs d'autres opérateurs peuvent être différents.

Questions concernant la documentation

Pour toute autre demande (suggestion, correction) concernant la documentation, envoyez une télécopie ou un courriel aux adresses suivantes :

Fax	+49 9131- 98 63315
E-mail	mailto:docu.motioncontrol@siemens.com

Adresse Internet de Siemens

Des informations actualisées en permanence sur les produits SIMOTION, le support produit et les FAQ sont à votre disposition sur Internet sous :

- Informations générales :
 - <http://www.siemens.de/simotion> (en allemand)
 - <http://www.siemens.com/simotion> (site international)
- Support produit :
 - <http://support.automation.siemens.com/WW/view/fr/10805436>

Autre assistance

Nous vous proposons des formations pour vous apprendre à travailler avec SIMOTION.

Veillez vous adresser au centre de formation de votre région ou au centre principal de formation qui se trouve en Allemagne, à D-90027 Nuremberg, Tél. : +49 (911) 895 3202.

Vous trouverez des informations sur l'offre de formation sur le site

www.sitrain.com

Sommaire

	Avant-propos	3
1	Description.....	9
1.1	Généralités.....	9
1.2	Mise en service	12
1.3	Insertion du moteur de positionnement SIMODRIVE POSMO A dans le projet SIMOTION	13
1.4	Intégration des blocs fonctionnels dans le projet d'utilisateur.....	14
1.5	Création de variables E/S	15
2	Blocs fonctionnels.....	17
2.1	Vue d'ensemble des blocs fonctionnels.....	17
2.2	Bloc fonctionnel _POSMOA_control	17
2.3	Bloc fonctionnel _POSMOA_nControl	24
2.4	Bloc fonctionnel _POSMOA_rwParameter	31
2.5	Bloc fonctionnel _POSMOA_rwAllParameter	35
2.6	Appel des blocs fonctionnels	42
3	Exemple d'application	45
3.1	Généralités sur l'exemple d'application	45
3.2	Commande et visualisation de l'exemple d'application dans l'affichage détaillé	47
3.3	Variables utilisées dans l'exemple d'application	52
A	Annexe	55
A.1	Descripteurs SIMOTION et SIMATIC	55
A.2	Liste des abréviations	60
	Index.....	61

Description

1.1 Généralités

Vue d'ensemble

SIMODRIVE POSMO A est un entraînement de positionnement décentralisé sur le bus de terrain PROFIBUS DP (esclave DP normalisé).

La partie puissance et le contrôle du mouvement sont intégrés dans le moteur.

Tous les signaux et données pour la mise en service et l'exploitation de l'entraînement sont transmis via le PROFIBUS DP.

L'énergie d'entraînement est fournie par une alimentation 24 V CC (moteur 75 W) ou par un raccordement 48 V CC (moteur 300 W).

La fonctionnalité intégrée de positionnement convient à un grand nombre d'applications monoaxe simples telles que le réglage de butées et de formats.

Remarque

Configuration matérielle / logicielle requise :

Les conditions suivantes s'appliquent aux fonctionnalités décrites dans ce manuel :

- Version matérielle POSMO A 75W : à partir de O
- Version logicielle POSMO A 75W : à partir de V3.0
- Version matérielle POSMO A 300W : à partir de G
- Version logicielle POSMO A 300W : à partir de V3.0

Les moteurs de positionnement POSMO A avec matériel et logiciel divergents peuvent être pilotés à l'aide des blocs fonctionnels intégrés à SIMOTION SCOUT V4.1. Cette fonctionnalité est limitée en fonction de la version matérielle/logicielle du moteur de positionnement POSMO A.

Conditions

Les fonctions standard décrites dans la présente documentation requièrent les versions logicielles suivantes :

- SIMOTION SCOUT V4.1 ou supérieure
- Noyau SIMOTION V4.1 ou supérieure
- Packages technologiques SIMOTION V4.1 ou supérieure

Communication

Le bus de terrain PROFIBUS DP permet un rapide échange de données cyclique entre l'esclave DP (POSMO A) et le maître DP de niveau supérieur (plate-forme matérielle SIMOTION, par ex. SIMOTION C2xx).

Informations supplémentaires

Remarque

Pour plus d'informations, reportez-vous au manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*, chapitre "Descriptif technique".

Cette documentation est fournie avec SIMOTION SCOUT en tant que document électronique.

Montage et raccordement

Pour les instructions et consignes de montage et de raccordement d'un SIMODRIVE POSMO A, reportez-vous au manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*, chapitre "Montage et raccordement".

Connectez le SIMODRIVE POSMO A à une des interfaces PROFIBUS DP sur l'appareil SIMOTION (plate-forme matérielle).

La figure suivante représente la connexion d'un entraînement SIMODRIVE POSMO A à un appareil SIMOTION (par ex. SIMOTION C2xx).

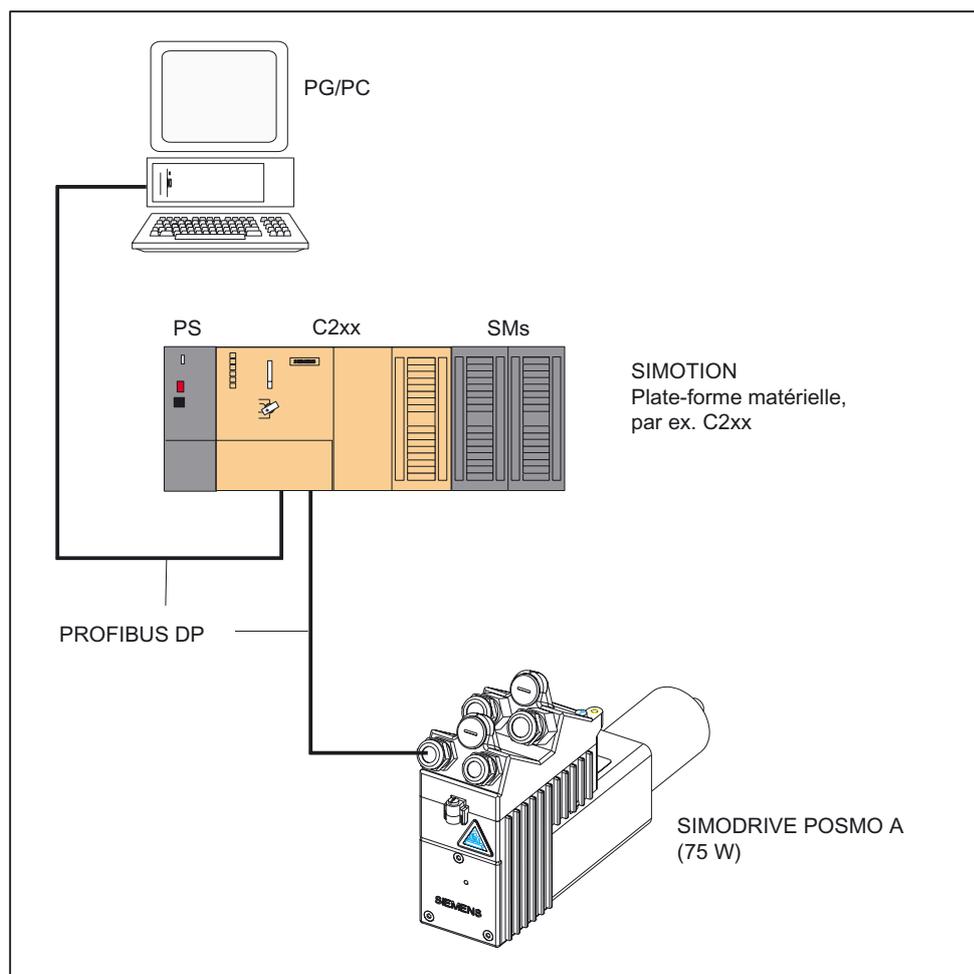


Figure 1-1 Connexion de SIMODRIVE POSMO A à l'appareil SIMOTION C2xx

1.2 Mise en service

Vue d'ensemble

Pour mettre en service le SIMODRIVE POSMO A et le commander à partir du système SIMOTION, effectuez les étapes suivantes :

1. Montez et câblez le moteur de positionnement SIMODRIVE POSMO A.
2. Réglez l'adresse réseau PROFIBUS DP sous le cache-bornes du SIMODRIVE POSMO A.
3. Activez la résistance de terminaison sur le premier et le dernier abonné.

Remarque

Pour les étapes 1 à 3, reportez-vous au manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*, chapitre "Montage et raccordement".

4. Le SIMODRIVE POSMO A peut être mis en service de la manière suivante :
 - à l'aide du maître C1 "SIMODRIVE POSMO A PROFIBUS MASTER"
 - à l'aide de l'outil de mise en service "SimoCom A"

Remarque

Voir le manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*, chapitre "Mise en service du maître DP".

- à l'aide de l'outil "Drive ES". Cet outil comporte "SimoCom A"

Remarque

Voir description fonctionnelle *Drive ES Basic*.

5. Insérez le SIMODRIVE POSMO A dans le projet SIMOTION (voir chapitre Insertion du moteur de positionnement SIMODRIVE POSMO A dans le projet SIMOTION (Page 13)).
6. Commande du SIMODRIVE POSMO A à partir du système SIMOTION à l'aide de blocs fonctionnels (voir chapitre Blocs fonctionnels (Page 17)).

Remarque

Les informations suivantes sont disponibles dans le manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP* :

- Mise en service de l'axe
 - Communication via PROFIBUS DP
 - Description des fonctions
 - Gestion des défauts et diagnostic
 - Montage et maintenance
-

1.3 Insertion du moteur de positionnement SIMODRIVE POSMO A dans le projet SIMOTION

Conditions

Les conditions suivantes doivent être remplies :

1. Un projet a été créé dans SIMOTION SCOUT et un châssis avec l'appareil SIMOTION a été inséré dans la configuration matérielle.
2. Un sous-réseau PROFIBUS a été configuré.

Remarque

Pour savoir comment créer un projet et configurer un sous-réseau PROFIBUS, reportez-vous à l'aide en ligne de SIMOTION SCOUT.

Insertion de SIMODRIVE POSMO A

Pour intégrer le SIMODRIVE POSMO A dans le sous-réseau PROFIBUS de votre projet, procédez comme suit :

1. Dans SIMOTION SCOUT, ouvrez la boîte de dialogue **Projet d'utilisateur** au moyen du menu **Projet > Ouvrir**. Dans cette boîte, sélectionnez votre projet, puis confirmez par **OK**.
2. Ouvrez **HW Config**(en double-cliquant sur l'appareil SIMOTION dans le navigateur de projet de SIMOTION SCOUT).
3. Dans la fenêtre **HW Config**, ouvrez le **catalogue matériel** à l'aide du menu **Affichage > Catalogue**.
4. Dans le catalogue matériel, ouvrez le dossier **PROFIBUS DP** , puis le sous-dossier **SIMODRIVE** et sélectionnez **SIMODRIVE POSMO A**.
5. Copiez le SIMODRIVE POSMO A vers le sous-réseau PROFIBUS de votre projet à l'aide d'un "Glisser-Déplacer"

La boîte de dialogue **Propriétés - Interface PROFIBUS SIMODRIVE POSMO A** s'ouvre. Sélectionnez l'adresse que vous avez réglée dans le cache-bornes du POSMO A (voir le manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP, chapitre "Montage et raccordement"*) et confirmez par **OK**.

Le moteur de positionnement SIMODRIVE POSMO A sélectionné sera inséré dans le projet.

6. Adresses E/S du POSMO A.

Lors de l'insertion du POSMO A dans votre projet SIMOTION, des valeurs par défaut sont affectées aux adresses E/S. Ces valeurs peuvent être affichées en sélectionnant le POSMO A inséré. Les adresses E/S peuvent être lues dans la partie inférieure de la fenêtre **HW Config**.

Avant l'appel des blocs fonctionnels, ces adresses doivent être créées en tant que variables E/S dans le navigateur de mnémoniques (voir chapitre Création de variables E/S (Page 15)).

1.4 Intégration des blocs fonctionnels dans le projet d'utilisateur

Création de l'instance des FB dans le projet d'utilisateur

Les blocs fonctionnels font partie intégrante de la bibliothèque de commandes du système d'ingénierie SIMOTION SCOUT. Pour la mise en oeuvre des blocs, une instance doit être créée dans le projet utilisateur pour chaque bloc fonctionnel utilisé et une variable de type **Struct_POSMOA_params** pour chaque utilisation du bloc fonctionnel **_POSMOA_rwAllParameter**.

Exemple :

```
VAR_GLOBAL
...
myPosmoAControl      : _POSMOA_control;          // FB for controlling of POSMO A
myPosmoArwParameter  : _POSMOA_rwParameter;     // FB for handling single parameter
myPosmoArwAllParameter : _POSMOA_rwAllParameter; // FB for handling parameterset
myAllParaPosmoA      : Struct_POSMOA_params;     // Variable for structure of all
                                                           // parameters POSMO A
...
END_VAR
```

Appel (représentation CONT)

Les descriptions respectives des blocs fonctionnels comportent une représentation CONT des blocs fonctionnels individuels.

Exemple d'application

Vous trouverez l'exemple d'application sur le CD-ROM "SIMOTION Utilities & Applications". L'exemple est disponible pour différentes plates-formes matérielles SIMOTION.

Le CD-ROM "SIMOTION Utilities & Applications" est joint gratuitement à la fourniture de SIMOTION SCOUT.

1.5 Création de variables E/S

Vue d'ensemble

La communication entre l'appareil SIMOTION et le SIMODRIVE POSMO A s'effectue à l'aide d'accès directs de périphérie. L'adressage des accès directs de périphérie en lecture/écriture s'effectue via les variables E/S.

Les noms des variables E/S peuvent être attribués arbitrairement dans SIMOTION SCOUT. Les variables E/S doivent être définies sous forme d'ARRAY [0..7] et [0..3] of BYTE. Attribuez-leur les adresses paramétrées dans la configuration matérielle.

Les noms des entrées de périphérie doivent être transférés aux blocs fonctionnels en tant que paramètres d'appel. Les données traitées pour les sorties de périphérie seront mises à disposition en tant que paramètre d'entrée / sortie du FB. Les paramètres d'entrée/sortie doivent être renseignés par des variables de type ARRAY [0..7] et [0..3] of BYTE. Après l'appel de bloc, ces variables E/S doivent être affectées aux sorties de périphérie (voir l'exemple d'appel du chapitre Appel des blocs fonctionnels (Page 42)).

Remarque

La variable affectée aux paramètres d'entrée / sortie ne doit pas être créée de manière temporaire (VAR_TEMP ou variable locale d'une fonction).

L'exemple suivant montre l'assignation des adresses de modules pour les variables E/S dans SIMOTION SCOUT.

	Nom	Adresse E/S	Lecture seule	Type de données	Longueur de champ
1	mypkwin	PIB 256		Array	8
2	mypzdin	PIB 264		Array	4
3	mypkwout	PQB 256	<input type="checkbox"/>	Array	8
4	mypzdout	PQB 264	<input type="checkbox"/>	Array	4

Figure 1-2 Affectation des adresses dans SIMOTION SCOUT

Les adresses d'entrée et de sortie possèdent chacune une plage de 8 octets (correspondant à la plage PKW du POSMO A) et une plage de 4 octets (correspondant à la plage PZD du POSMO A).

Remarque

Pour plus d'informations, voir :

- Aide en ligne de *SIMOTION SCOUT*
- Manuel de programmation du langage de programmation correspondant, par ex. :
 - le manuel de programmation *SIMOTION ST, Structured Text*
 - le manuel de programmation *SIMOTION MCC, Motion Control Chart*
 - le manuel de programmation *SIMOTION KOP/FUP, Schéma à contacts et logigramme*

Ces documents sont fournis avec SIMOTION SCOUT au format électronique !

Blocs fonctionnels

2.1 Vue d'ensemble des blocs fonctionnels

Ce chapitre contient une description de tous les blocs fonctionnels (FB) et la structure de données requise pour la communication entre un appareil SIMOTION et le SIMODRIVE POSMO A.

Les blocs fonctionnels forment l'interface logicielle entre le système SIMOTION et le moteur de positionnement SIMODRIVE POSMO A.

L'utilisation de ces blocs fonctionnels facilite la commande et le paramétrage du moteur de positionnement SIMODRIVE POSMO A à partir du programme SIMOTION.

Il est possible, par exemple, de paramétrer un POSMO A sans connaître les formats des paramètres PROFIBUS ni les identificateurs de requête.

Les blocs fonctionnels doivent être appelés plusieurs fois (de façon cyclique) à partir du programme utilisateur.

Les blocs fonctionnels sont les suivants :

- Bloc fonctionnel `_POSMOA_control` (Page 17)
- Bloc fonctionnel `_POSMOA_nControl` (Page 24) (à partir de la version 4.1)
- Bloc fonctionnel `_POSMOA_rwParameter` (Page 31)
- Bloc fonctionnel `_POSMOA_rwAllParameter` (Page 35)

Remarque

Pour la commande totale et la communication du SIMODRIVE POSMO A à partir du programme SIMOTION, il faut créer une instance des blocs fonctionnels `_POSMOA_rwParameter`, `_POSMOA_rwAllParameter` et suivant le mode de fonctionnement paramétré (vitesse ou positionneur), une instance des blocs fonctionnels `_POSMOA_control` ou `_POSMOA_nControl`.

Remarque

Si le SIMODRIVE POSMO A est déconnecté, puis reconnecté à l'alimentation de puissance, les données de déplacement MDI transmises auparavant doivent de nouveau être transmises au POSMO A (à ce sujet, voir le tableau "Paramètre du FB `_POSMOA_control`").

2.2 Bloc fonctionnel `_POSMOA_control`

Tâche

Les blocs fonctionnels `_POSMOA_control` permettent de piloter le SIMODRIVE POSMO A connecté.

Les fonctions sont les suivantes :

- Initialiser

Met l'entraînement à l'état "prêt à fonctionner".

Conditions :

- Aucun défaut n'est signalé par l'entraînement (**driveError** = FALSE)
- L'acquiescement des défauts n'est pas activé (**resetError** = FALSE)

- Référencement

Définit le point de référence de l'entraînement.

- Marche par à-coups

L'entraînement fonctionne en asservissement de vitesse dans le sens positif ou négatif.

- Exécution de programme

Démarre, arrête ou annule un bloc adressé au moyen de **blockNumber** ou bien un bloc spécifique à l'intérieur d'un programme.

- MDI

Déplace l'entraînement avec la vitesse et l'accélération spécifiée vers une position définie.

Les paramètres MDI sont transférés dans le bloc 3.

Le bloc MDI peut être démarré avec **blockNumber** = 3 et **start** = TRUE.

- Acquiescement des défauts

Acquiesce une erreur dans l'entraînement.

Remarque

L'entraînement ne peut être déplacé qu'après l'acquiescement d'une erreur. Le paramètre **enable** doit contenir la valeur TRUE.

- Mode automatique bloc par bloc / Mode automatique

Les signalisations en retour sont les suivantes :

- Bloc de déplacement actuel
- Prêt à fonctionner
- Informations concernant les alarmes et les défauts
- Etat complet (mot d'état et octet de réponse)
- Etat de la transmission de données

Tableau 2- 1 Paramètre du FB _POSMA_control

Nom	Type P 1)	Type de données	Default	Signification
pkwIn	IN	ARRAY [0..7] of BYTE	8(16#00)	Transfert des entrées de périphérie du POSMA A au FB
pzdIn	IN	ARRAY [0..3] of BYTE	4(16#00)	Transfert des entrées de périphérie du POSMA A au FB
enable	IN	BOOL	FALSE	Met l'entraînement à l'état "prêt à fonctionner". En l'absence de défaut, l'entraînement est prêt à fonctionner après cet ordre.
homing	IN	BOOL	FALSE	Positionne le point de référence Ce signal doit être présent pendant au moins 50 ms.
releaseBrake ⁴⁾	IN	BOOL	FALSE	= TRUE : Desserrer le frein de maintien = FALSE : Commande séquentielle de freinage active
jog1	IN	BOOL	FALSE	Sélectionne Marche par à-coups 1 Si les paramètres Marche par à-coups 1 et 2 sont activés en même temps, une alarme est générée et l'entraînement est arrêté.
jog2	IN	BOOL	FALSE	Sélectionne Marche par à-coups 2 Si les paramètres Marche par à-coups 1 et 2 sont activés en même temps, une alarme est générée et l'entraînement est arrêté.
jogOverride ³⁾	IN	INT	20	Correction de vitesse Marche par à-coups (0...100 %) La correction de vitesse peut également être modifiée en cours d'exécution.
veloOverride ³⁾	IN	INT	20	Correction de vitesse (0...100 %) La correction de vitesse peut également être modifiée en cours de déplacement.
start	IN	BOOL	FALSE	= front FALSE → TRUE : Le bloc de déplacement spécifié sous blockNumber est démarré. Après sélection d'un bloc dans blockNumber , le paramètre start ne peut être activé qu'à l'appel de bloc suivant.
singleBlock ⁴⁾	IN	BOOL	FALSE	= TRUE : Automatique Bloc par bloc, chaque bloc doit être redémarré. = FALSE : Mode automatique
enableRdIn ⁴⁾	IN	BOOL	TRUE	= TRUE : Déblocage chargement, le bloc suivant est débloqué pour l'exécution. = FALSE : blocage du chargement.
extBlockChange ⁴⁾	IN	BOOL	FALSE	= front FALSE → TRUE : interruption du bloc actif et passage au bloc suivant. = FALSE : Pas de changement de bloc externe.
noStopIntermediate	IN	BOOL	FALSE	= TRUE : aucun arrêt intermédiaire ou reprise de l'exécution d'un bloc sur un arrêt intermédiaire = FALSE : Arrêt intermédiaire Interruption de la commande de mouvement en cours, start non accepté

Nom	Type P ¹⁾	Type de données	Default	Signification
noStop	IN	BOOL	FALSE	= TRUE : aucun arrêt = FALSE : arrêt annulation de la commande de mouvement actuelle. Lorsqu'un paramètre start est activé simultanément, start n'est pas accepté.
blockNumber	IN	BYTE	16#00	Numéro de déplacement 3...27 bloc par bloc ou programme blockNumber = 3 → mode MDI
resetError	IN	BOOL	FALSE	Acquitte un défaut 1. Supprime la cause du défaut. 2. Front FALSE → TRUE 3. Le paramètre doit rester TRUE jusqu'à ce que driveError = FALSE.
setStartInformation	IN	BYTE	16#00	Octet de départ Combinaison d'octets qui est transmise en tant que condition de démarrage supplémentaire. ²⁾
mdiMode ³⁾	IN	BOOL	FALSE	= TRUE : MDI relatif La valeur du paramètre mdiPosition est évaluée en fonction de la position actuelle = FALSE : MDI absolu La valeur du paramètre mdiPosition est évaluée de manière absolue en fonction de la position zéro de l'entraînement définie par la prise de référence.
mdiVelocity ³⁾	IN	INT	0	Vitesse du déplacement MDI (0...100 %)
mdiAcceleration ³⁾	IN	INT	0	Accélération du déplacement MDI (0...100 %)
mdiPosition ³⁾	IN	REAL	0	Position de destination du déplacement MDI Plage de valeurs : $-2 \cdot 10^5$ à $2 \cdot 10^5$
reqControl ⁴⁾	IN	BOOL	FALSE	Maîtrise demandée par la commande. p701 = 1: Substitution de télégramme active. = TRUE : Les données PROFIBUS sont reprises par POSMA A. = FALSE : Les données du PROFIBUS sont gelées, les dernières données reçues sont utilisées. p701 = 0: Substitution de télégramme inactive, comportement identique au POSMA A de version logicielle < V3.0.
pkwOut	IN/OUT	ARRAY [0..7] of BYTE	-	Données du FB traitées pour les sorties de périphérie du POSMA A
pzdOut	IN/OUT	ARRAY [0..3] of BYTE	-	Données du FB traitées pour les sorties de périphérie du POSMA A
busy	IN/OUT	BOOL	-	Coordination des FB
ready	OUT	BOOL	FALSE	Entraînement prêt à fonctionner Affectation : Mot d'état bit 2...0 ²⁾
active	OUT	BOOL	FALSE	= TRUE : Axe en mouvement

Nom	Type P ¹⁾	Type de données	Default	Signification
dataReady ⁴⁾	OUT	BOOL	FALSE	Pour la transmission, par exemple mdiPosition , jogOverride , plusieurs cycles sont nécessaires. = TRUE : Transmission de données terminée (par exemple mdiPosition , jogOverride ,...) = FALSE : Transmission de données en cours (temps de montée) ou transmission de données pas encore déclenchée
statusWord	OUT	WORD	16#0000	Affichage du mot d'état ²⁾
actBlockNumber	OUT	BYTE	16#00	Affichage du numéro de bloc actuel
statusInformation	OUT	BYTE	16#00	Octet de réponse Combinaison d'octets en tant que signal d'état supplémentaire. ²⁾
driveWarning	OUT	BOOL	FALSE	Présence d'une alarme de l'entraînement (voir paramètre driveWarnId).
driveWarnId	OUT	WORD	16#0000	Cause de l'alarme Format d'octet Cette valeur correspond au paramètre 953 (alarmes) ²⁾
driveWarnInfo ⁴⁾	OUT	WORD	16#0000	Alarmes ou informations complémentaires, correspond au p954 du POSMO A
driveError	OUT	BOOL	FALSE	Présence d'un défaut de l'entraînement (voir paramètre driveErrorId).
driveErrorId	OUT	WORD	16#0000	Cause du défaut Format d'octet Cette valeur correspond au paramètre 947 (défauts) ²⁾
error	OUT	BOOL	FALSE	= TRUE : requête terminée avec erreur (voir paramètre errorID)
errorID	OUT	WORD	16#0000	Numéro de l'erreur de paramétrage signalée par l'entraînement (page PKW) ²⁾

¹⁾ Types de paramètres : IN = paramètre d'entrée, OUT = paramètre de sortie, IN/OUT = paramètre d'entrée/sortie

²⁾ voir manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*

³⁾ Ce paramètre est uniquement transmis lors de la modification de la valeur du paramètre.

⁴⁾ Ce paramètre fait partie du FB **_POSMOA_control** à partir de SIMOTION V4.1 et peut uniquement être exploité avec POSMO A à partir de la version logicielle V3.0.

Substitution de télégramme (POS MO A version logicielle V3.0 et supérieure)

Pour certaines applications, l'entraînement ne doit en aucun cas s'arrêter involontairement ou il doit être possible de configurer un "gel" de l'état de l'entraînement pour la mise à l'arrêt du maître (appareil SIMOTION).

La fonction "Substitution de télégramme" peut être activée avec le paramètre d'entrée **reqControl** à partir de la version logicielle V3.0 du POS MO A si le paramètre **p701 = TRUE**.

Les données PZD envoyées par l'appareil SIMOTION sont reprises par le POS MO A à l'aide de **reqControl = TRUE**. Avec le passage de **TRUE** à **FALSE** du paramètre d'entrée **reqControl**, le POS MO A utilise les données PZD dernièrement reçues (STW, sélection de bloc et octet de départ). Si le paramètre **P701 = FALSE**, l'état du paramètre d'entrée **reqControl** n'est pas exploité.

Remarque

La fonction "Substitution de télégramme" est immédiatement active en réglant **p701 = 1**.

Veillez à ce que le moteur puisse à tout moment être arrêté par un **ARRET D'URGENCE**.

Pour plus d'informations, consultez le manuel de l'utilisateur SIMODRIVE POS MO A *Moteur de positionnement décentralisé sur PROFIBUS DP*.

Intégration dans une tâche (appel)

Le bloc fonctionnel **_POS MOA_control** doit être appelé de manière cyclique dans **BackgroundTask** ou dans **TimerInterruptTask**. Un appel dans **SystemInterruptTask** n'est pas admis. Un appel du FB dans des tâches synchrones (**IPOSynchronousTask**) n'est pas recommandé en raison des durées d'exécution.

Remarque

La fonctionnalité du FB **_POS MOA_Control** a été étendue dans la version 4.1. Afin de pouvoir utiliser la nouvelle fonctionnalité, vous devez compléter les nouveaux paramètres d'entrée lors de l'appel du FB **_POS MOA_Control**. Lorsque vous souhaitez travailler avec la fonctionnalité antérieure (< V4.1), vous pouvez omettre les nouveaux paramètres d'entrée dans l'appel du FB en notation détaillée.

Signalisations d'erreurs, défauts et alarmes

La valeur **TRUE** du paramètre de sortie **error** signale une erreur de paramétrage de l'ALM. Le paramètre de sortie **errorID** fournit des informations détaillées concernant l'erreur de paramétrage décelée ou signalée par le POS MO A. Les erreurs de paramétrage ne doivent pas nécessairement être acquittées. Les paramètres modifiés (par exemple, temps de montée) peuvent être retransmis. Les défauts dans POS MO A sont signalés dans le paramètre de sortie **driveError** avec la valeur **TRUE**. La raison du défaut peut être lue dans le paramètre de sortie **driveError** (la valeur correspond à P947). Les défauts de l'entraînement doivent être acquittés et être réinitialisés par un front montant sur le paramètre d'entrée **resetError**.

Les alarmes en présence du POS MO A et les informations correspondantes sont émises dans les paramètres de sortie **driveWarning**, **driveWarnId** (valeur correspondant à P953), **driveWarnInfo** (valeur correspondant à P954).

2.3 Bloc fonctionnel _POSMOA_nControl

Tâche

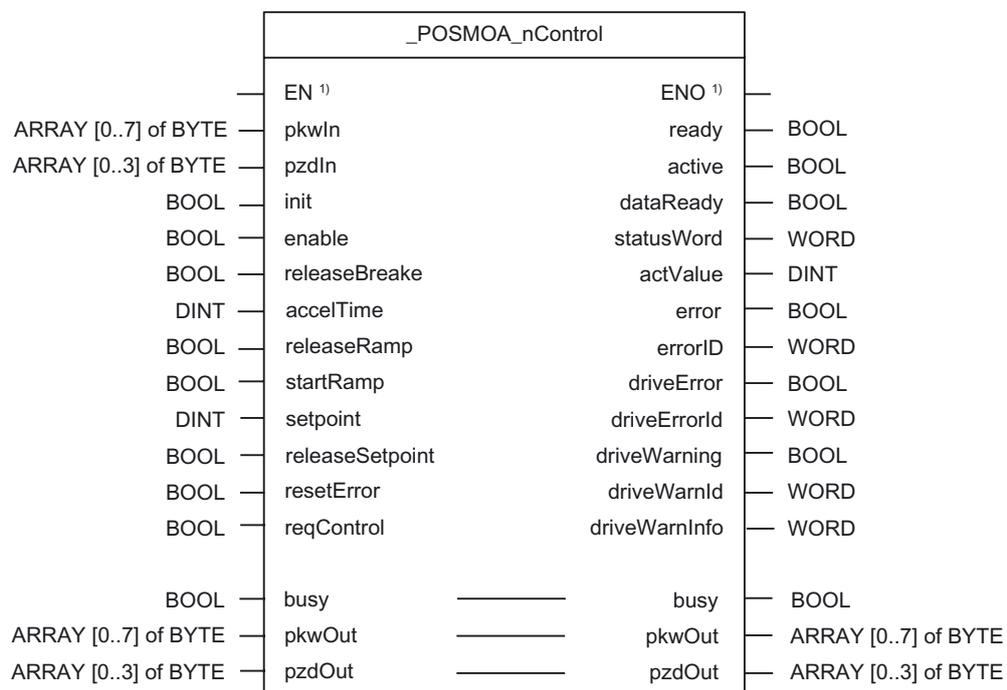
Le bloc fonctionnel **_POSMOA_ncontrol** permet de commander le SIMODRIVE POSMO A en régulation de vitesse.

Remarque

Le bloc fonctionnel **_POSMOA_nControl** est contenu dans SIMOTION SCOUT V4.1 et supérieure.

Le POSMO A ne peut pas être exploité via les objets technologiques (TO) au moyen d'une interface de consigne de vitesse.

Appel (représentation CONT)



1) paramètre spécifique à CONT

Description des paramètres

Remarque

Le paramètre **busy** ne doit **pas** être écrasé par l'utilisateur. Ce paramètre est renseigné et interrogé par le bloc fonctionnel. Il suffit de le renseigner lors de l'appel du FB correspondant avec une variable globale créée par l'utilisateur. Ce paramètre assure la coordination des différents FB pour le POSMO A. Ce mécanisme empêche l'accès simultané de plusieurs FB à un POSMO A.

Tableau 2- 2 Paramètres du FB _POSMA_nControl

Nom	Type P ¹⁾	Type de données	Default	Signification
pkwIn	IN	ARRAY[0..7] of BYTE	8(16#00)	Transfert des entrées de périphérie du POSMO A au FB _POSMA_nControl
pzdIn	IN	ARRAY[0..3] of BYTE	4(16#00)	Transfert des entrées de périphérie du POSMO A au FB
init	IN	BOOL	FALSE	= TRUE : met l'entraînement à l'état prêt à l'enclenchement STW = 0x040E
enable	IN	BOOL	FALSE	= TRUE : met l'entraînement à l'état prêt à fonctionner En l'absence de défaut, l'entraînement est prêt à fonctionner après cet ordre.
releaseBrake	IN	BOOL	FALSE	= TRUE : Desserrer le frein de maintien = FALSE : Commande séquentielle de freinage active
accelTime	IN	DINT	0	Temps de montée/descente [ms] En régulation de vitesse, la valeur de consigne évolue comme suit durant ce laps de temps : <ul style="list-style-type: none"> • Démarrage : de zéro à la vitesse de rotation maximale autorisée • Descente : de la vitesse de rotation maximale autorisée à zéro
releaseRamp	IN	BOOL	FALSE	= TRUE : Déblocage de la sortie du générateur de rampe
startRamp	IN	BOOL	FALSE	= front FALSE → TRUE : Démarrer généré de rampe
setpoint	IN	INT	0	Consigne de vitesse
releaseSetpoint	IN	BOOL	FALSE	Libération de la consigne = TRUE: Consigne libérée
resetError	IN	BOOL	FALSE	Acquitte un défaut <ol style="list-style-type: none"> 1. Supprime la cause du défaut. 2. Front FALSE → TRUE 3. Le paramètre doit rester TRUE jusqu'à ce que driveError = FALSE.
reqControl	IN	BOOL	FALSE	Maîtrise demandée par la commande. p701 = 1: Substitution de télégramme active. = TRUE : Les données PROFIBUS sont reprises par le POSMO A. = FALSE : Les données du PROFIBUS sont gelées, les dernières données reçues sont utilisées. p701=0 : Substitution de télégramme inactive, comportement identique au POSMO A de version logicielle < V3.0.
busy	IN/OUT	BOOL	-	Coordination des blocs fonctionnels
pkwOut	IN/OUT	ARRAY[0..7] of BYTE	-	Données du FB traitées pour les sorties de périphérie du POSMO A (interface PKW)
pzdOut	IN/OUT	ARRAY[0..3] of BYTE	-	Données du FB traitées pour les sorties de périphérie du POSMO A (interface PZD)
ready	OUT	BOOL	FALSE	Entraînement prêt à fonctionner, opération ET : mot d'état bit 2, bit 1, bit 0
active	OUT	BOOL	FALSE	= TRUE : Moteur en fonctionnement (n > 0)

Nom	Type P ¹⁾	Type de données	Default	Signification
dataReady	OUT	BOOL	FALSE	Pour la transmission, par exemple du temps de montée, plusieurs cycles sont nécessaires. La fin de la transmission de données est signalée par un front montant. = TRUE : Transmission de données prête, les données sont transmises. = FALSE : Transmission de données en cours (par exemple temps de montée)
statusWord	OUT	WORD	16#0000	Affichage du mot d'état
actValue	OUT	DINT	0	Mesure de vitesse
error	OUT	BOOL	FALSE	= TRUE : requête terminée avec erreur (voir paramètre errorID)
errorID	OUT	WORD	16#0000	Numéro de l'erreur de paramétrage signalée par l'entraînement (plage PKW) ²⁾
driveError	OUT	BOOL	FALSE	Présence d'un défaut dans l'entraînement
driveErrorId	OUT	WORD	16#0000	Cause du défaut Format d'octet Cette valeur correspond au paramètre 947 (défauts) ²⁾
driveWarning	OUT	BOOL	FALSE	Présence d'une alarme de l'entraînement (voir paramètre driveWarnId).
driveWarnId	OUT	WORD	16#0000	Cause de l'alarme Format d'octet Cette valeur correspond au paramètre 953 (alarmes) ²⁾
driveWarnInfo	OUT	WORD	16#0000	Alarmes, informations complémentaires (pour le POSMA A à partir de la version de firmware 1.4) Format d'octet LA valeur correspond à P954, (alarmes, informations complémentaires)

1) Types de paramètres : IN = paramètre d'entrée, OUT = paramètre de sortie, IN/OUT = paramètre d'entrée/sortie

2) voir manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*

Description des fonctions

Avec le niveau **TRUE** sur le paramètre d'entrée **init**, le POSMO A passe à l'état "prêt à l'enclenchement" (mot de commande 0x040E). L'état "prêt à l'enclenchement" est signalé dans le paramètre de sortie **statusWord** Bit 0 = **TRUE**. Avec le passage du niveau **FALSE** à **TRUE** sur le paramètre d'entrée **enable**, l'entraînement passe à l'état prêt à fonctionner. Le POSMO A passe à l'état "prêt à fonctionner", ce qui est signalé par le paramètre de sortie **ready** = **TRUE**. Avec le paramètre d'entrée **enableSetpoint** = **TRUE** (libération de la consigne), **enableRamp** = **TRUE** (déblocage du générateur de rampe), un front montant sur le paramètre d'entrée **execRamp** = **TRUE** (démarrage du générateur de rampe) et **setpoint** > 0 (valeur de consigne), le mouvement de déplacement démarre.

L'ordre chronologique pour le réglage des paramètres d'entrée **enable**, **enableSetpoint**, **enableRamp**, **execRamp** en vue de démarrer le mouvement de déplacement est indifférent. Les paramètres d'entrée mentionnés ont le même rang de priorité. La réinitialisation d'un des paramètres d'entrée **enable**, **enableSetpoint**, **enableRamp**, **execRamp** provoque l'arrêt du mouvement de déplacement.

La transmission des paramètres (par exemple temps de montée, paramètre d'entrée **accelTime**) nécessite plusieurs cycles de tâche. Si une nouvelle valeur est attribuée au paramètre d'entrée **accelTime**, le paramètre de sortie **dataReady** prend la valeur **FALSE**. Les erreurs de paramétrage éventuellement présentes (paramètre de sortie **error** = **TRUE**) sont réinitialisées. Lorsque le paramètre d'entrée/sortie **busy** prend la valeur **FALSE**, la transmission des données est effectuée immédiatement. Si une transmission de données (par ex. lecture de paramètres par FB _POSMA_rwParameter) est active au moment d'un nouveau paramétrage du paramètre d'entrée **accelTime**, la transmission de données est ajournée jusqu'à ce que **busy** = **FALSE**. La valeur qui est paramétrée sur le paramètre d'entrée **accelTime** Time au moment du passage de **busy** = **TRUE** à **busy** = **FALSE** est transmise. La fin de la transmission de données est signalée par un front montant sur le paramètre de sortie **dataReady** qui conserve alors la valeur **TRUE** jusqu'à la prochaine transmission de données.

Substitution de télégramme (POSMA A version logicielle V3.0 et supérieure)

Pour certaines applications, l'entraînement ne doit en aucun cas s'arrêter involontairement ou il doit être possible de configurer un "gel" de l'état de l'entraînement pour la mise à l'arrêt du maître (appareil SIMOTION).

La fonction "Substitution de télégramme" peut être activée avec le paramètre d'entrée **reqControl** à partir de la version logicielle V3.0 du POSMO A si le paramètre **p701** = **TRUE**.

Les données PZD envoyées par l'appareil SIMOTION sont reprises par le POSMO A à l'aide de **reqControl** = **TRUE**. Avec le passage de **TRUE** à **FALSE** du paramètre d'entrée **reqControl**, le POSMO A utilise les données PZD dernièrement reçues (STW, sélection de bloc et octet de départ). Si le paramètre **P701** = **FALSE**, l'état du paramètre d'entrée **reqControl** n'est pas évalué.

Remarque

La fonction "Substitution de télégramme" est immédiatement active en réglant **p701** = 1.

Veillez à ce que le moteur puisse à tout moment être arrêté par un ARRET D'URGENCE.

Pour plus d'informations, consultez le manuel de l'utilisateur SIMODRIVE POSMO A *Moteur de positionnement décentralisé sur PROFIBUS DP*.

Vue d'ensemble graphique des fonctions

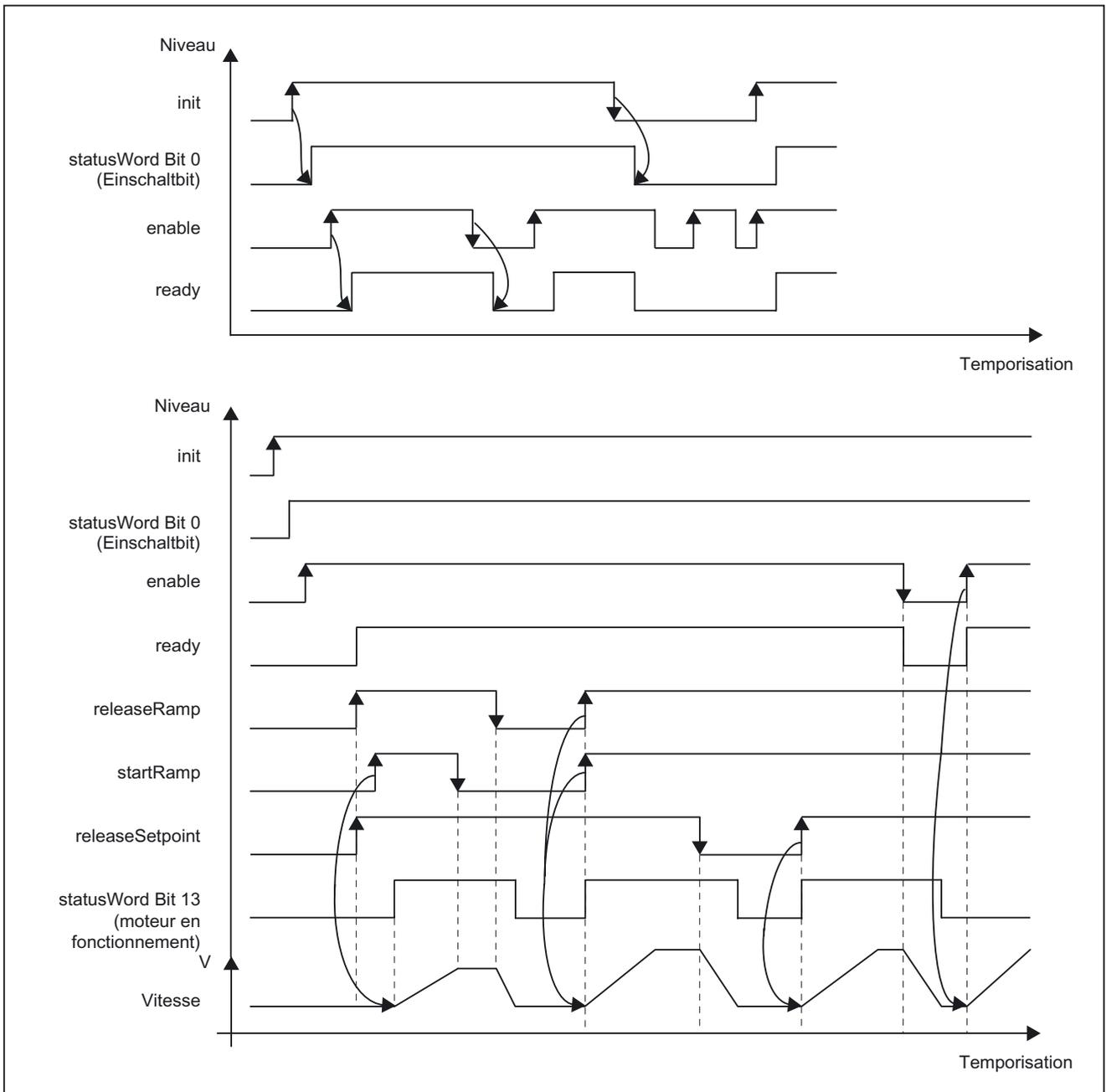


Figure 2-1 Diagramme de transfert des signaux

Intégration dans une tâche (appel)

Le bloc fonctionnel `_POSMOA_nControl` doit être appelé de manière cyclique dans **BackgroundTask** ou dans **TimerInterruptTask**. Un appel dans **SystemInterruptTask** n'est pas admis. Un appel du FB dans des tâches synchrones (**IPOSynchronousTask**) n'est pas recommandé en raison des durées d'exécution.

Signalisations d'erreurs, défauts et alarmes

La valeur **TRUE** du paramètre de sortie **error** signale une erreur de paramétrage de l'ALM. Le paramètre de sortie **errorID** fournit des informations détaillées sur les erreurs de paramétrage survenues. Les erreurs de paramétrage ne doivent pas nécessairement être acquittées. Les paramètres modifiés (par exemple, temps de montée) peuvent être retransmis.

Les défauts dans POSMO A sont signalés dans le paramètre de sortie **driveError** avec la valeur **TRUE**. La raison du défaut peut être lue dans le paramètre de sortie **driveError** (la valeur correspond à P947). Les défauts de l'entraînement doivent être acquittés et être réinitialisés par un front montant sur le paramètre d'entrée **resetError**.

Les alarmes en présence du POSMO A et les informations correspondantes sont émises dans les paramètres de sortie **driveWarning**, **driveWarnId** (valeur correspondant à P953), **driveWarnInfo** (valeur correspondant à P954).

2.4 Bloc fonctionnel _POSMAO_rwParameter

Tâche

Les blocs fonctionnels **_POSMAO_rwParameter** permettent de paramétrer le SIMODRIVE POSMO A connecté.

Les fonctions sont les suivantes :

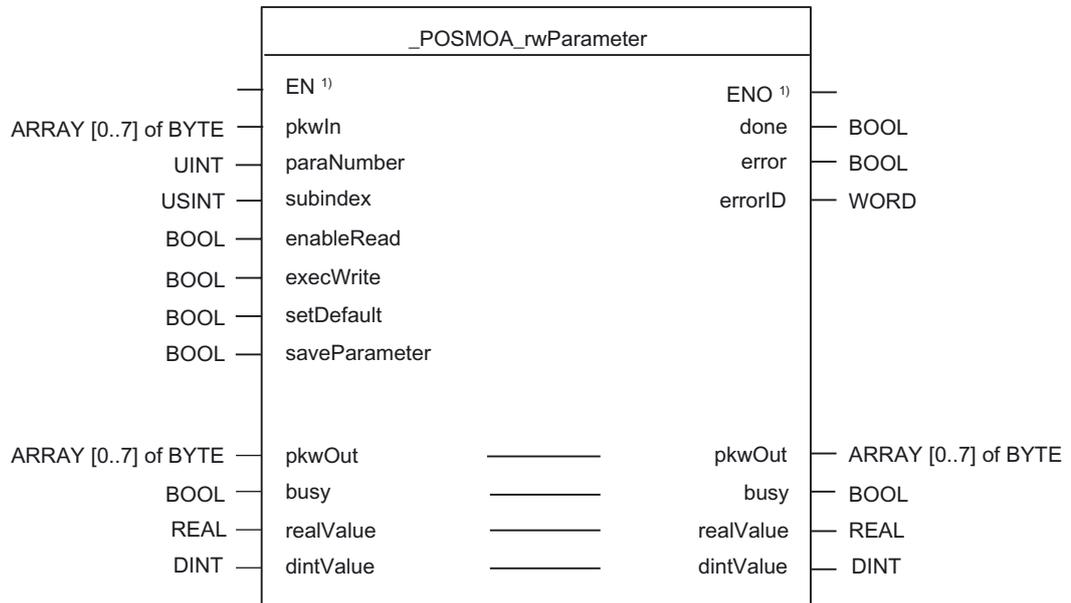
- Lire le paramètre : Fournit la valeur du paramètre indiqué.
- Ecrire la valeur du paramètre : Règle le paramètre indiqué à la valeur saisie.
- Charger le préréglage usine : Réinitialise la configuration des paramètres aux préréglages usine.
- Enregistrement des paramètres : Sauvegarde la configuration de paramètres actuelle dans la mémoire non volatile.

Les paramètres suivants peuvent être lus / écrits par les blocs fonctionnels :

Numéro de paramètre	lecture à partir du POSMO A	écriture dans le POSMO A (mode positionnement)	écriture dans le POSMO A (mode régulation de vitesse)
1...5	oui	oui	oui
6...7	oui	oui	non
9...23	oui	oui	oui
24	oui	oui	non
25...38	oui	oui	oui
39...53	oui	non	non
54	oui	oui	oui
55	oui	non	non
56...61	oui	oui	oui
62	oui	non	non
80:28...87:28	oui	oui	oui
99:21	oui	oui	oui
100	oui	oui	non
101:11	oui	oui	non
700 ¹⁾	oui	oui	oui
701 ¹⁾	oui	oui	oui
880 ¹⁾	oui	oui	oui
918...928	non	non	non
930	oui	non	non
947...954	non	non	non
964:8	oui	non	non
967...990:78	non	non	non
1426 ¹⁾	oui	oui	oui
1427 ¹⁾	oui	oui	oui

¹⁾ Ce paramètre est nouveau ou étendu avec SIMOTION V4.1.

Appel (représentation CONT)



1) Paramètres spécifiques CONT

Description des paramètres

Remarque

Les descripteurs SIMOTION ont été modifiés à partir de la version V4.0.

Vous trouverez une comparaison des descripteurs SIMOTION et SIMATIC dans l'annexe Descripteurs SIMOTION et SIMATIC (Page 55).

Le paramètre **busy** ne doit **pas** être écrasé par l'utilisateur. Ce paramètre est renseigné et interrogé par le bloc fonctionnel. Il suffit de le renseigner lors de l'appel du FB correspondant avec une variable globale créée par l'utilisateur. Ce paramètre assure la coordination des différents FB pour le POSMO A. Ce mécanisme empêche l'accès simultané de plusieurs FB à un POSMO A.

Tableau 2- 3 Paramètre du FB _POSMAO_rwParameter

Nom	Type P ¹⁾	Type de données	Default	Signification
pkwIn	IN	ARRAY [0..7] of BYTE	8(16#00)	Transfert des entrées de périphérie du POSMO A au FB
paraNumber	IN	UINT	0	Numéro du paramètre devant être lu ou écrit
subindex	IN	USINT	0	Subindex mettre à 0 dans le cas des paramètres sans indice Dans le cas des paramètres de type tableau, cette valeur est l'indice de tableau. ²⁾
enableRead	IN	BOOL	FALSE	= TRUE : lire le paramètre de manière cyclique = front FALSE → TRUE : lire le paramètre de manière unique
execWrite	IN	BOOL	FALSE	= front FALSE → TRUE : écrire le paramètre Si ce paramètre est activé en même temps que enableRead , la lecture est exécutée.
setDefault	IN	BOOL	FALSE	= front FALSE → TRUE : charger le pré réglage usine Si ce paramètre est activé en même temps que enableRead , la lecture est exécutée.
saveParameter	IN	BOOL	FALSE	= front FALSE → TRUE : sauvegarder les paramètres Si ce paramètre est activé en même temps que enableRead , la lecture est exécutée.
pkwOut	IN/OUT	ARRAY [0..7] of BYTE	-	Transfert des données traitées du FB aux sorties de périphérie du POSMO A
busy	IN/OUT	BOOL	-	Coordination des FB
realValue	IN/OUT	REAL	-	écriture → pour la valeur à écrire (type de données C4 et N2) ²⁾ lecture → pour la valeur à lire (type de données C4 et N2) ²⁾
dintValue	IN/OUT	DINT	-	écriture → pour la valeur à écrire (type de données I2, T2, V2 et T4) ²⁾ lecture → pour la valeur à lire (type de données I2, T2, V2 et T4) ²⁾
done	OUT	BOOL	FALSE	= TRUE : lorsque la tâche actuelle est terminée = FALSE : aucune tâche n'est présente ou bien une tâche est en cours de traitement
error	OUT	BOOL	FALSE	= TRUE : tâche terminée avec erreur (voir paramètre errorID)
errorID	OUT	WORD	16#0000	Numéro de l'erreur de paramétrage signalée par l'entraînement (plage PKW) ²⁾

1) Types de paramètres : IN = paramètre d'entrée, OUT = paramètre de sortie, IN/OUT = paramètre d'entrée/sortie

2) voir manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*

Intégration dans une tâche (appel)

Le bloc fonctionnel `_POSMOA_rwParameter` doit être appelé de manière cyclique dans **BackgroundTask** ou dans **TimerInterruptTask**. Un appel dans **SystemInterruptTask** n'est pas admis. Un appel du FB dans des tâches synchrones (**IPOSynchronousTask**) n'est pas recommandé en raison des durées d'exécution.

Signalisations d'erreurs

La valeur **TRUE** du paramètre de sortie **error** signale une erreur de paramétrage. Le paramètre de sortie **errorID** fournit des informations détaillées concernant l'erreur de paramétrage décelée ou signalée par le POSMO A. Les erreurs de paramétrage ne doivent pas nécessairement être acquittées. Les paramètres modifiés (par exemple, temps de montée) peuvent être retransmis.

2.5 Bloc fonctionnel _POSMAO_rwAllParameter

Tâche

Le bloc fonctionnel **_POSMAO_rwAllParameter** permet de lire et d'écrire le jeu de paramètres du SIMODRIVE POSMO A connecté.

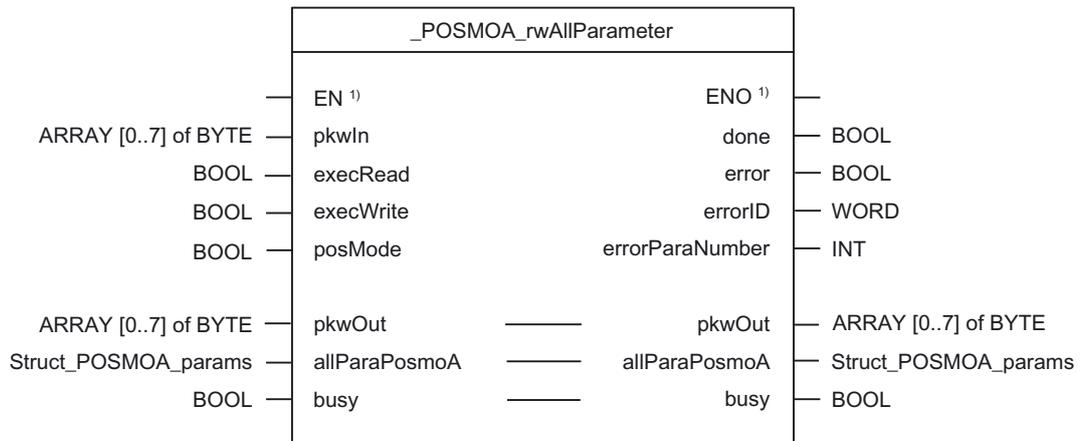
Lors de l'appel de l'instance de FB correspondante, les données à lire et à écrire sont sauvegardées dans une variable créée par l'utilisateur avec la structure de données **Struct_POSMAO_params**.

Les paramètres suivants sont lus / écrits par les blocs fonctionnels :

Número de paramètre	lecture à partir du POSMO A	écriture dans le POSMO A (mode positionnement)	écriture dans le POSMO A (mode régulation de vitesse)
1...5	oui	oui	oui
6...7	oui	oui	non
9...23	oui	oui	oui
24	oui	oui	non
25...38	oui	oui	oui
39...53	oui	non	non
54	oui	oui	oui
55	oui	non	non
56...61	oui	oui	oui
62	oui	non	non
80:28...87:28	oui	oui	oui
99:21	oui	oui	oui
100	oui	oui	non
101:11	oui	oui	non
700 ¹⁾	oui	oui	oui
701 ¹⁾	oui	oui	oui
880 ¹⁾	oui	oui	oui
918...928	non	non	non
930	oui	non	non
947...954	non	non	non
964:8	oui	non	non
967...990:116	non	non	non
1426 ¹⁾	oui	oui	oui
1427 ¹⁾	oui	oui	oui

¹⁾ Ce paramètre est nouveau ou étendu avec SIMOTION V4.1.

Appel (représentation CONT)



1) paramètre spécifique à CONT

Description des paramètres

Remarque

Les descripteurs SIMOTION ont été modifiés à partir de la version V4.0.

Vous trouverez une comparaison des descripteurs SIMOTION et SIMATIC dans l'annexe Descripteurs SIMOTION et SIMATIC (Page 55).

Le paramètre **busy** ne doit **pas** être écrasé par l'utilisateur. Ce paramètre est renseigné et interrogé par le bloc fonctionnel. Il suffit de le renseigner lors de l'appel du FB correspondant avec une variable globale créée par l'utilisateur. Ce paramètre assure la coordination des différents FB pour le POSMO A. Ce mécanisme empêche l'accès simultané de plusieurs FB à un POSMO A.

Tableau 2- 4 Paramètre du FB _POSMAO_rwAllParameter

Nom	Type P ¹⁾	Type de données	Default	Signification
pkwIn	IN	ARRAY [0..7] of BYTE	8(16#00)	Transfert des entrées de périphérie du POSMO A au FB
execRead	IN	BOOL	FALSE	= front FALSE → TRUE : lire toutes les données de manière unique Le déclenchement est effectué par un front positif.
execWrite	IN	BOOL	FALSE	= front FALSE → TRUE : écrire toutes les données de manière unique Le déclenchement est effectué par un front positif.
posMode ³⁾	IN	BOOL	TRUE	= TRUE : Mode positionnement du POSMO A = FALSE : fonctionnement en asservissement de vitesse
pkwOut	IN/OUT	ARRAY [0..7] of BYTE	-	Données du FB traitées pour les sorties de périphérie du POSMO A
allParaPosmoA	IN/OUT	Struct_POSMOA_params	-	Structure de données pour tous les paramètres du POSMO A
busy	IN/OUT	BOOL	-	Coordination des FB
done	OUT	BOOL	FALSE	= TRUE : lorsque la tâche actuelle est terminée = FALSE : aucune tâche n'est présente ou bien une tâche est en cours de traitement
error	OUT	BOOL	FALSE	= TRUE : tâche terminée avec erreur (voir paramètre errorID)
errorID	OUT	WORD	16#0000	Numéro de l'erreur de paramétrage signalée par l'entraînement (plage PKW) ²⁾
errorParaNumber	OUT	INT	0	Numéro du paramètre ayant causé le défaut ²⁾

1) Types de paramètres : IN = paramètre d'entrée, OUT = paramètre de sortie, IN/OUT = paramètre d'entrée/sortie

2) voir manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*

3) Ce paramètre fait partie du FB **_POSMAO_rwAllParameter** à partir de SIMOTION V4.1 et peut être exploité uniquement avec POSMO A à partir de la version logicielle 3.0.

Structure de données Struct_POSMOA_params

La structure de données de type **Struct_POSMOA_params** contient tous les paramètres du SIMODRIVE POSMO A.

La structure de données est utilisée par le bloc fonctionnel **_POSMAO_rwAllParameter**. L'accès aux éléments de la structure de données s'effectue via la variable autodéfinie du type de données **Struct_POSMOA_params**.

Le tableau suivant contient la structure de données **Struct_POSMOA_params**.

Remarque

Les descripteurs SIMOTION ont été modifiés à partir de la version V4.0.

Vous trouverez une comparaison des descripteurs SIMOTION et SIMATIC dans l'annexe Descripteurs SIMOTION et SIMATIC (Page 55).

Tableau 2- 5 Structure de données Struct_POSMAO_params

Nom	type	Valeur initiale	Commentaire	r/w ¹⁾
p1	REAL	0.0	Axe linéaire/rotatif	r/w
p2	REAL	10.0	Course par tour de réducteur	r/w
p3	REAL	1.0	Rapport de transmission	r/w
p4	INT	0	Unité	r/w
p5	REAL	0.0	Position du point de référence	r/w
p6	REAL	-200000.0	Fin de course logiciel Début	r/w
p7	REAL	200000.0	Fin de course logiciel Fin	r/w
p8	REAL	3000.0	Vitesse maximale	r/w
p9	INT	10	Temps de montée	r/w
p10	DINT	30000	Vitesse maximale	r/w
p11	REAL	2.0	Zone de destination	r/w
p12	REAL	20000.0	Ecart de traînage maximal	r/w
p13	DINT	50	Temps de surveillance	r/w
p14	REAL	20000.0	Fenêtre d'immobilisation	r/w
p15	REAL	0.0	Compensation du jeu à l'inversion	r/w
p16	REAL	9.0	Surintensité maximale	r/w
p17	DINT	20	Gain P Régulateur de vitesse	r/w
p18	INT	22	Temps d'intégration Régulateur de vitesse	r/w
p19	REAL	1.0	Gain de boucle	r/w
p20	REAL	30.0	Lissage de la consigne de courant	r/w
p21	REAL	2.0	Lissage de la consigne de vitesse	r/w
p22	REAL	1000.0	Accélération maximale	r/w
p23	DINT	0	Constante de temps A-coups	r/w
p24	INT	100	Fonction Override	r/w
p25	INT	100	Correction Accélération	r/w
p26	INT	20	Correction Vitesse Marche par à-coups	r/w
p27	INT	50	Correction Accélération Marche par à-coups	r/w
p28	REAL	9.0	Courant maximal	r/w
p29	DINT	12000	Température de l'électronique Temps de tolérance	r/w
p30	INT	0	Suppression des perturbations	r/w
p31	INT	0	Fonction Borne 1	r/w
p32	INT	0	Fonction Borne 2	r/w
p33	DINT	0	Adresse Sortie de mesure 1	r/w
p34	INT	7	Facteur Shift Sortie de mesure 1	r/w
p35	INT	128	Offset Sortie de mesure 1	r/w
p36	DINT	0	Adresse Sortie de mesure 2	r/w
p37	INT	0	Facteur Shift Sortie de mesure 2	r/w
p38	INT	128	Offset Sortie de mesure 2	r/w
p39	REAL	0.0	Consigne de position	r
p40	REAL	0.0	Mesure de position	r
p41	REAL	0.0	Consigne de vitesse	r

Nom	type	Valeur initiale	Commentaire	r/w ¹⁾
p42	REAL	0.0	Mesure de vitesse	r
p43	REAL	0.0	Consigne de courant	r
p44	REAL	0.0	Mesure de courant	r
p45	DINT	0	Valeur de temporisation	r
p46	REAL	0.0	Erreur de traînage	r
p47	REAL	0.0	Température de l'électronique	r
p48	INT	0	Numéro de déplacement actuel	r
p49	INT	0	Numéro de bloc suivant	r
p50	DINT	0	Consigne de vitesse	r
p51	DINT	0	Mesure de vitesse	r
p52	DINT	0	Version du matériel	r
p53	DINT	0	Version du firmware	r
p54	DINT	5	Gain P Régulateur de vitesse Immobilisation	r/w
p55	REAL	0.0	Position de signal	r
p56	INT	0	Position de fonctionnement	r/w
p57	DINT	20	Gain P Régulateur de maintien Immobilisation (à partir de la version matérielle F)	r/w
p58	DINT	100	Frein de maintien Temps d'ouverture de frein	r/w
p59	REAL	10.0	Vitesse Serrage du frein de maintien	r/w
p60	DINT	400	Frein de maintien Temps de décélération en freinage	r/w
p61	DINT	100	Frein de maintien Temps de blocage du régulateur	r/w
p62	REAL	0.0	Position de mesure	r
p80	ARRAY[0..27] of Array_POSMOA_prgCtrlInfo		Blocs de déplacement 1 à 27 voir tableau "Structure Array_POSMOA_prgCtrlInfo"	r/w
p81	ARRAY[0..27] of REAL	28(0.0)	Position de destination Blocs de déplacement 1 à 27	r/w
p82	ARRAY[0..27] of INT	28(100)	Vitesse ou Vitesse de rotation Blocs de déplacement 1 à 27	r/w
p83	ARRAY[0..27] of INT	28(100)	Accélération Blocs de déplacement 1 à 27	r/w
p84	ARRAY[0..27] of DINT	28(0)	Valeur de temporisation Blocs de déplacement 1 à 27	r/w
p85	ARRAY[0..27] of REAL	28(0.0)	Position de signalisation Blocs de déplacement 1 à 27	r/w
p86	ARRAY[0..27] of INT	28(0)	SMStart MMStart Blocs de déplacements 1 à 27	r/w
p87	ARRAY[0..27] of INT	28(0)	MMStop MMPos Blocs de déplacement 1 à 27	r/w
p99	ARRAY[0..20] of INT	13,18,23,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	Gestion de programmes (voir manuel de l'utilisateur <i>Moteur de positionnement décentralisé sur PROFIBUS DP</i>)	r/w
p100	INT	0	Simulation du mot de commande	r/w
p101	ARRAY[0..10] of INT	10(0)	Blocs 1 à 10 de la structure de données pour les paramètre de POSMO A	r/w

Nom	type	Valeur initiale	Commentaire	r/w ¹⁾
p700 ²⁾	INT	2	Service 1 = mode régulation de vitesse 2 = mode positionnement	r/w
p701 ²⁾	INT	0	Substitution de télégramme	r/w
p880 ²⁾	REAL	4096	Normalisation de la vitesse de rotation à la sortie du réducteur, lorsqu'une consigne de 4096 a été spécifiée sous forme décimale par le mot de commande (STW)	r/w
p930	INT	0	Mode de fonctionnement actuel 1 = mode régulation de vitesse 2 = mode positionnement	r
p964	ARRAY[0..7] of INT	8(0)	Identification de l'entraînement	r
p1426 ²⁾	REAL	100	Bande de tolérance, mesure de vitesse	r/w
p1427 ²⁾	INT	0	Temporisation du signal "démarrage terminé"	r/w

¹⁾ r - lire, w - écrire

²⁾ Ce paramètre est nouveau ou étendu avec SIMOTION V4.1.

Structure "Array_POSMAO_prgCtrlInfo"

"Array_POSMAO_prgCtrlInfo" contient le mot de commande de programme. Ce tableau permet de déterminer le comportement d'un bloc de déplacement (voir manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*).

Tableau 2- 6 Structure Array_POSMAO_prgCtrlInfo

Elément de tableau	Type de données	Valeur initiale	Commentaire
0	BOOL	TRUE	Type de mouvement
1	BOOL	TRUE	Type de positionnement
2	BOOL	FALSE	Type de temporisation
3	BOOL	FALSE	Combinaison logique entre temporisation et octet de départ
4	BOOL	FALSE	Retour au programme
5	BOOL	FALSE	Type de déplacement
6	BOOL	FALSE	Inverser Condition d'octet de départ
7	BOOL	FALSE	Type SMStart
8	BOOL	FALSE	Programme Arrêt
9	BOOL	FALSE	Forcer la position réelle

Intégration dans une tâche (appel)

Le bloc fonctionnel `_POSMOA_rwAllParameter` doit être appelé de manière cyclique dans **BackgroundTask** ou dans **TimerInterruptTask**. Un appel dans **SystemInterruptTask** n'est pas admis. Un appel du FB dans des tâches synchrones (**IPOSynchronousTask**) n'est pas recommandé en raison des durées d'exécution.

Remarque

La fonctionnalité du FB `_POSMOA_rwAllParameter` a été ajoutée à la V4.1. Afin de pouvoir utiliser la nouvelle fonctionnalité, vous devez compléter le nouveau paramètre d'entrée **posMode** lors de l'appel du FB `_POSMOA_rwAllParameter`.

Lorsque vous souhaitez travailler avec la fonctionnalité antérieure (< V4.1), vous pouvez omettre le nouveau paramètre d'entrée dans l'appel du FB en notation détaillée.

Signalisations d'erreurs

La valeur **TRUE** du paramètre de sortie **error** signale une erreur de paramétrage de l'ALM. Le paramètre de sortie **errorID** fournit des informations détaillées concernant l'erreur de paramétrage décelée ou signalée par le POSMO A. Le paramètre de sortie **errorParaNumber** fournit le numéro du paramètre qui a déclenché l'erreur.

Les erreurs de paramétrage ne doivent pas nécessairement être acquittées. Les paramètres modifiés (par exemple, temps de montée) peuvent être retransmis.

2.6 Appel des blocs fonctionnels

Pour pouvoir utiliser les blocs fonctionnels dans votre programme applicatif, procédez comme suit (les chiffres représentés dans l'extrait de programme qui suit correspondent aux points de l'énumération suivante) :

1. Créer l'instance du bloc fonctionnel (voir extrait de programme ci-dessous, par ex. créer une instance pour le FB `_POSMOA_control`).
2. Créez la variable pour la structure de données (seulement pour le FB `_POSMOA_rwAllParameter`).
3. Créez le champ pour le paramètre d'entrée / sortie.
4. Appelez l'instance du bloc fonctionnel.
5. Transmettez le paramètre d'entrée.
6. L'accès aux paramètres de sortie du FB s'effectue à l'aide de `<Nom d'instance du FB>.<Nom du paramètre de sortie>`.
7. Les données traitées par le FB pour les sorties de périphérie sont affectées aux variables E/S par le champ créé dans le point 3 (voir l'extrait de programme suivant).

Remarque

Si vous souhaitez piloter plusieurs SIMODRIVE POSMO A, vous devez, pour chaque POSMO A mis en oeuvre, créer une nouvelle variable pour la structure de données (FB `_POSMOA_rwAllParameter`) ainsi que des instances de FB avec de nouveaux noms.

Exemple d'appel

```
UNIT E_posmoA;

INTERFACE

// Definition of global variables for demo program
VAR_GLOBAL
  myPosmoAControl  : _POSMA_control;    // create "_POSMA_control" instance      (1)
  myEnable         : BOOL;              // enable posmoA
  myHoming         : BOOL;              // homing posmoA
  myJogPositive    : BOOL;              // jog positive posmoA
  myJogNegative    : BOOL;              // jog negative posmoA
  myBusy           : BOOL;              // coordination bit
  myError          : BOOL;              // variable created by user for accessing
                                          // an output variable of the function block

END_VAR

PROGRAM ExamplePosmoA;                  // Program of BackgroundTask
END_INTERFACE

IMPLEMENTATION
PROGRAM ExamplePosmoA                    // Program of BackgroundTask
  VAR
    // temporary array for outputs of FBs      (3)
    tmpPkwOutput : ARRAY[0..7] of BYTE;
    tmpPzdOutput : ARRAY[0..3] of BYTE;
  END_VAR
  // INSTANCE CALL of FB _POSMA_control      (4)
  myPosmoAControl (   pkwIn   := myPkwIn,      (5)
                     pzdIn   := myPzdIn,
                     enable  := myEnable,
                     homing  := myHoming,
                     jog1    := myJogNegative,
                     jog2    := myJogPositive,
                     busy    := myBusy,
                     pkwOut  := tmpPkwOutput,
                     pzdOut  := tmpPzdOutput
                    );
  // an output variable in the "_POSMA_control" function block is assigned to a      (6)
  // "myError" variable created by the user.
  myError := myPosmoAControl.error;

  // Assignment of intermediate buffer byte arrays to I/O addresses      (7)
  myPkwOut := tmpPkwOutput;
  myPzdOut := tmpPzdOutput;

END_PROGRAM                               // ExamplePosmoA
END_IMPLEMENTATION
```

Remarque

Le PROGRAMME ExamplePosmoA doit être assigné dans le système exclusif.

Exemple d'application

3.1 Généralités sur l'exemple d'application

Tâche

L'exemple d'application vous montre comment commander le POSMO A à l'aide des blocs fonctionnels et comment lire et écrire les paramètres d'entraînement du POSMO A.

Il existe pour cela une interface de commande **enumCommands** permettant de démarrer l'action désirée (par ex. la marche par à-coups).

La structure de données **Struct_checkbacks** indique les états des actions ainsi que d'autres informations.

Les modes de fonctionnement ou fonctionnalité suivants sont mis en oeuvre :

- Référencement
Variante : "accoster le repère visuel avec l'axe et affecter la mesure"
- Marche par à-coups
Déplacement dans le sens positif ou négatif
- MDI
Déplacement vers la position souhaitée
- Manipulation de paramètres
 - Lire ou écrire des paramètres individuels
 - Sauvegarder tous les paramètres dans l'EEPROM du POSMO A
- Lecture de la position réelle actuelle
- Pour la marche par à-coups et pour MDI, la position réelle actuelle du POSMO A est lue de manière cyclique et enregistrée dans une variable.

Plate-forme matérielle

L'exemple d'application est disponible pour différentes plates-formes matérielles SIMOTION.

Remarque

Lorsque l'exemple d'application n'est pas disponible pour votre plate-forme matérielle SIMOTION, vous devez adapter la configuration matérielle.

Adaptation de l'exemple d'application

La configuration de l'exemple et le matériel dont vous disposez doivent être adaptés.

Vous avez les possibilités suivantes :

1. Adapter la configuration de l'exemple au matériel disponible (mise en service de l'entraînement, adresse PROFIBUS-DP).
2. Adapter la configuration du matériel disponible à l'exemple (mise en service de l'entraînement, adresse PROFIBUS-DP).

Remarque

Pour la mise en service de l'entraînement, tenir compte de la documentation de l'entraînement. Cette documentation est fournie avec SIMOTION SCOUT en tant que document électronique.

Appel de l'exemple d'application

L'exemple d'application se trouve sur le CD-ROM "SIMOTION Utilities & Applications" Le CD-ROM "SIMOTION Utilities & Applications" est fourni gratuitement avec SIMOTION SCOUT.

1. Désarchiver et ouvrir le projet avec l'exemple d'application.
2. Vérifiez la configuration matérielle : Adresses PROFIBUS DP.
3. Enregistrez et compilez le projet d'exemple. Vous pouvez alors charger l'exemple dans l'appareil SIMOTION et régler l'état de fonctionnement **RUN**.

Les autres étapes de manipulation de l'exemple sont effectuées dans le navigateur de mnémoniques à l'aide des Enums sous **myCommand**. Pour ce faire, l'élément "E_posmoA" doit être sélectionné dans le répertoire **PROGRAMMES** du navigateur de projet. Cliquez sur **Forçage immédiat** pour affecter les valeurs de la colonne "Valeur de forçage" aux variables correspondantes.

Signalisations d'erreurs

Les erreurs et alarmes présentes (par ex. en mode marche par à-coups ou lors de la lecture d'un paramètre) sont affichées dans les variables suivantes :

- myCheckbacks.error = TRUE
Une erreur s'est produite (tâche annulée ; un défaut est présent sur le POSMO A).
- myCheckbacks.ctrlErrorID
Spécification d'erreur du bloc fonctionnel **_POSMOA_control**
Numéro de l'erreur de paramétrage signalée par le POSMO A
- myCheckbacks.driveErrorID
Spécification d'erreur du POSMO A
Cause d'un défaut signalé par le POSMO A
- myCheckbacks.rwErrorID
Spécification d'erreur du bloc fonctionnel **_POSMOA_rwParameter**
Une erreur s'est produite lors de la lecture ou de l'écriture.
- myCheckbacks.driveWarning = TRUE
Une alarme est présente sur le POSMO A.
- myCheckbacks.driveWarningID = TRUE
Numéro d'une alarme signalée par le POSMO A

3.2 Commande et visualisation de l'exemple d'application dans l'affichage détaillé

Lors d'une transition STOP-RUN de votre appareil SIMOTION, le POSMO A est initialisé automatiquement. L'état "prêt à fonctionner" de l'entraînement est signalé en retour par le POSMO A via les variables suivantes :

- myCheckbacks.driveReady = TRUE

Sélectionnez le mode de fonctionnement

L'utilisateur a le choix entre les modes de fonctionnement "Marche par à-coups", "Référencement", "MDI" et "Manipulation de paramètres". La variable **myCommand** permet de sélectionner le mode de fonctionnement.

Mode "Marche par à-coups" :

Le mode de fonctionnement "Marche par à-coups" permet de déplacer le POSMO A dans le sens positif et négatif. La "Marche par à-coups" s'effectue au moyen des réglages de paramètres suivants dans l'instance du bloc fonctionnel **_POSMOA_control** créée :

- jogOverride = 100
- veloOverride = 100
- noStopIntermediate = TRUE Pas d'arrêt intermédiaire
- noStop = TRUE aucun arrêt

Activez les cases à cocher des variables suivantes dans la colonne "Valeur de forçage" du navigateur de mnémoniques et sélectionnez la valeur à associer.

- myCommand = START_JOG_POSITIVE Marche par à-coups en sens positif
- myCommand = START_JOG_NEGATIVE Marche par à-coups en sens négatif

En cliquant sur **Forçage immédiat**, la valeur est affectée aux variables et le POSMO A se déplace dans le sens correspondant. Pour lire l'état actuel du mode "Marche par à-coups" dans le navigateur de mnémoniques, procédez comme suit :

- myCheckbacks.actCommand = JOG_POSITIVE_ACTIVE
Marche par à-coups activée en sens positif
- myCheckbacks.actCommand = JOG_NEGATIVE_ACTIVE
Marche par à-coups en sens négatif activée
- myCheckbacks.jogPositiveBusy = TRUE
Le POSMO A se déplace dans le sens positif.
- myCheckbacks.jogNegativeBusy = TRUE
Le POSMO A se déplace dans le sens négatif.

La position réelle actuelle du POSMO A peut être lue dans le navigateur de mnémoniques, dans la variable **myCheckbacks.actPosition**.

Remarque

Le mode de fonctionnement "Marche par à-coups" ne peut être quitté qu'après l'arrêt du POSMO A (myCommand = STOP).

Mode "Référencement"

La variante "accoster le repère visuel avec l'axe et affecter la mesure" a été mise en oeuvre. Lors du référencement, le POSMO A doit être en mode régulation et à l'arrêt. La mesure (paramètre 40 du POSMO A) peut être activée via le mode de fonctionnement "Manipulation de paramètres" > "Lire des paramètres individuels".

Activez les cases à cocher des variables suivantes dans la colonne "Valeur de forçage" du navigateur de mnémoniques et sélectionnez la valeur à associer.

- myCommand = START_HOMING

En cliquant sur **Forçage immédiat**, la valeur est affectée aux variables et le POSMO A est référencé sur la valeur indiquée dans le paramètre 40 du POSMO A.

Remarque

Pour plus d'informations sur le référencement du POSMO A, consultez le manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*.

Cette documentation est fournie avec SIMOTION SCOUT en tant que document électronique.

Pour lire l'état actuel du mode "Référencement" dans le navigateur de mnémoniques, procédez comme suit :

- myCheckbacks.actCommand = HOMING_ACTIVE
Référencement activé
- myCheckbacks.actCommand = NO_COMMAND_ACTIVE
myCheckbacks.done = TRUE
Référencement terminé sans défaut

Mode "MDI"

Condition : Le POSMO A doit être référencé.

Dans le mode de fonctionnement "MDI", un bloc MDI peut être exécuté de manière absolue sur le POSMO A.

L'exécution du bloc MDI s'effectue au moyen des réglages de paramètres suivants dans l'instance du bloc fonctionnel **_POSMOA_control** créée :

- mdiMode = FALSE MDI absolu
- mdiVelocity = 100
- mdiAcceleration = 100
- veloOverride = 100
- noStopIntermediate = TRUE Pas d'arrêt intermédiaire
- noStop = TRUE pas d'arrêt

La position de destination du bloc MDI doit être spécifiée dans la variable **myAbsolutePosition**.

Activez les cases à cocher des variables suivantes dans la colonne "Valeur de forçage" du navigateur de mnémoniques et sélectionnez la valeur à associer.

- myCommand = START_MDI_BLOCK_ABSOLUTE
- myAbsolutePosition = ... Spécifiez la position de destination du bloc MDI.
(Valeur par défaut = 0,0)

En cliquant sur **Forçage immédiat**, les valeurs sont affectées aux variables et le POSMO A exécute le bloc MDI de manière absolue.

Pour lire l'état actuel de la "MDI" dans le navigateur de mnémoniques, procédez comme suit :

- myCheckbacks.actCommand = MDI_BLOCK_ACTIVE
Déplacement bloc MDI absolu activé
- myCheckbacks.actCommand = NO_COMMAND_ACTIVE
- myCheckbacks.done = TRUE
- myCheckbacks.positionReached = TRUE
Bloc MDI sans erreur de déplacement

La position réelle actuelle du POSMO A peut être lue dans le navigateur de mnémoniques, dans la variable **myCheckbacks.actPosition**.

Remarque

Le mode de fonctionnement "MDI" ne peut être quitté qu'après arrêt du POSMO A (myCommand = STOP).

Mode "Manipulation de paramètres"

Le mode de fonctionnement "Manipulation de paramètres" permet de lire et d'écrire des paramètres individuels et de sauvegarder tous les paramètres dans l'EEPROM du POSMO A.

Lecture de paramètres individuels

Le paramètre que vous souhaitez lire doit être spécifié dans la variable **myRdParaNumber**. Le sous-indice du paramètre que vous souhaitez lire doit être spécifié dans la variable **myRdSubIndex** (seulement pour les paramètres indexés). Activez les cases à cocher des variables suivantes dans la colonne "Valeur de forçage" du navigateur de mnémoniques et sélectionnez la valeur à associer.

- myCommand = READ_ONE_PARAMETER
Lecture de paramètres individuels
- myRdParaNumber = ...
Vous saisissez ici le numéro du paramètre à lire.
- myRdSubIndex = ...
Vous saisissez ici le sous-indice du paramètre à lire (seulement pour les paramètres indexés)

La valeur lue est enregistrée dans la variable **myReadValue**.

Écriture de paramètres individuels

Le paramètre que vous souhaitez écrire doit être spécifié dans la variable **myWrParaNumber**. Le sous-indice du paramètre que vous souhaitez écrire doit être spécifié dans la variable **myWrSubIndex** (seulement pour les paramètres indexés). Spécifiez la valeur souhaitée du paramètre à écrire dans la variable **myWrRealValue** (types de données C4 et N2) ²⁾ ou **myWrDintValue** (types de données I2, T2, V2 et T4) ²⁾. Activez les cases à cocher des variables suivantes dans la colonne "Valeur de forçage" du navigateur de mnémoniques et sélectionnez la valeur à associer.

- myCommand = WRITE_ONE_PARAMETER
Écriture de paramètres individuels
- myWrParaNumber = ...
Vous saisissez ici le numéro du paramètre à écrire.
- myWrSubIndex = ...
Vous saisissez ici le sous-indice du paramètre à écrire (seulement pour les paramètres indexés)
- myWrRealValue = ...
Vous saisissez ici la valeur du paramètre à écrire (type de données C4 et N2) ²⁾.
- myWrDintValue = ...
Vous saisissez ici la valeur du paramètre à écrire (types de données I2, T2, V2 et T4) ²⁾.

²⁾ Consultez le manuel de l'utilisateur *Moteur de positionnement décentralisé sur PROFIBUS DP*.

Cette documentation est jointe au format électronique à SIMOTION SCOUT.

Enregistrement de tous les paramètres dans l'EEPROM

Activez les cases à cocher des variables suivantes dans la colonne "Valeur de forçage" du navigateur de mnémoniques et sélectionnez la valeur à associer. Cliquez sur **Forçage immédiat** pour affecter la valeur à la variable et enregistrer tous les paramètres dans l'EEPROM.

- myCommand = SAVE_PARAMETER Enregistrement de tous les paramètres dans l'EEPROM

Pour lire l'état actuel de "Manipulation de paramètres" dans le navigateur de mnémoniques, procédez comme suit :

- myCheckbacks.actCommand = READ_PARA_ACTIVE
Lecture de paramètres individuels activée
- myCheckbacks.actCommand = WRITE_PARA_ACTIVE
Ecriture de paramètres individuels activée
- myCheckbacks.actCommand = SAVE_PARAMETER_ACTIVE
Vérifiez que tous les paramètres de POSMO A sont activés
- myCheckbacks.actCommand = NO_COMMAND_ACTIVE
myCheckbacks.done = TRUE
Manipulation de paramètres sans erreur terminée

Acquittement de défauts sur le POSMO A

Pour acquitter des défauts sur le POSMO A, procédez comme suit :

- myCommand = RESET_ERRORS

Pour lire l'état actuel d'acquittement des défauts dans le navigateur de mnémoniques, procédez comme suit :

- myCheckbacks.actCommand = RESET_ERRORS_ACTIVE
Acquittement des défauts activé
- myCheckbacks.actCommand = NO_COMMAND_ACTIVE
myCheckbacks.done = TRUE
Acquittement des défauts terminé

Remarque

Un défaut sur le POSMO A ne peut être acquitté avec succès que si la cause du défaut a été supprimée.

3.3 Variables utilisées dans l'exemple d'application

Tableau 3- 1 Vue d'ensemble des variables utilisées

Mnémonique	Type de données	Valeur initiale	Signification
myCommand	enumCommands	NO_COMMAND	Interface de commande
myCheckbacks	Struct_Checkbacks	NO_COMMAND_ACTIVE	Etat des commandes Informations complémentaires

Tableau 3- 2 Vue d'ensemble des Enums enumCommands

Mnémonique	Valeur d'Enum	Signification
START_HOMING	0	Démarrage référencement du POSMO A
START_JOG_NEGATIVE	1	Démarrage Marche par à-coups en sens négatif
START_JOG_POSITIVE	2	Démarrage Marche par à-coups en sens positif
START_MDI_BLOCK_ABSOLUTE	3	Démarrage Exécution du bloc MDI Mode absolu
STOP	4	Arrêt de toutes les actions
RESET_ERRORS	5	Acquittement des défauts sur le POSMO A
READ_ONE_PARAMETER	6	Démarrage lecture de paramètres individuels
WRITE_ONE_PARAMETER	7	Démarrage écriture de paramètres individuels
SAVE_PARAMETER	8	Sauvegarde de tous les paramètres dans l'EEPROM du POSMO A
NO_COMMAND	9	N'exécuter aucune action

Tableau 3- 3 Vue d'ensemble des Enums enumActCommand

Mnémonique	Valeur d'Enum	Signification
HOMING_ACTIVE	0	Référencement du POSMO A activé
JOG_NEGATIVE_ACTIVE	1	Marche par à-coups en sens négatif activée
JOG_POSITIVE_ACTIVE	2	Marche par à-coups activée en sens positif
MDI_BLOCK_ACTIVE	3	Exécution du bloc MDI activée
STOP_ACTIVE	4	Arrêt de toutes les actions activé
RESET_ERRORS_ACTIVE	5	Acquittement des défauts sur le POSMO A activé
READ_PARA_ACTIVE	6	Lecture de paramètres individuels activée
WRITE_PARA_ACTIVE	7	Ecriture de paramètres individuels activée
SAVE_PARAMETER_ACTIVE	8	Sauvegarde de tous les paramètres dans l'EEPROM du POSMO A activée
NO_COMMAND_ACTIVE	9	Aucune action activée

Tableau 3- 4 Structure de données Struct_checkbacks

Mnémonique	Type de données	Signification
actCommand	enumActCommand	Enums pour actions activées
done	BOOL	Action terminée
driveReady	BOOL	Entraînement (POSMA A) en mode "prêt à fonctionner"
jogPositiveBusy	BOOL	Marche par à-coups en sens positif activée
jogNegativeBusy	BOOL	Marche par à-coups en sens négatif activée
actPosition	REAL	Position réelle actuelle du POSMA A
positionReached	BOOL	Position atteinte lors de l'exécution absolue du bloc MDI
error	BOOL	Une erreur s'est produite (tâche annulée ; un défaut est présent sur le POSMA A)
ctrlErrorID	WORD	Spécification d'erreur du bloc _POSMAA_control Numéro de l'erreur de paramètre signalée par le POSMA A
driveErrorID	WORD	Spécification d'erreur du POSMA A Cause d'un défaut signalé par le POSMA A
rwErrorID	WORD	Spécification d'erreur du bloc _POSMAA_rwParameter Une erreur s'est produite lors de la lecture ou de l'écriture
driveWarning	BOOL	Une alarme est présente sur le POSMA A
driveWarningID	WORD	Numéro d'alarme

Annexe

A.1 Descripteurs SIMOTION et SIMATIC

Le tableau suivant contient une comparaison des descripteurs SIMOTION et SIMATIC.

Tableau A- 1 Descripteurs SIMOTION et SIMATIC SIMODRIVE POSMO A

Nom utilisé dans le système SIMOTION V4.1 ou supérieure (bibliothèque d'instructions de SCOUT)	Nom dans le système SIMATIC	Nom dans le système SIMOTION jusqu'à la version V3.2 (SIMOTION Function Library)
Paramètres des blocs fonctionnels		
_POSMOA_control	FB 10	_FB_posmoA_control
pkwIn	I_O_address	PKWInputInterface
pzdIn	I_O_address	PZDInputInterface
enable	Initialization	initialize
homing	Referencing	homing
releaseBrake	Brake_release	-
jog1	Jogging_1	jog1
jog2	Jogging_2	jog2
jogOverride	Jogging_override	jogOverride
start	Start	start
singleBlock	Automatic_operation	-
enableRdIn	Read_in_enable	-
extBlockChange	External_blockchange	-
noStopIntermediate	No_intermediate_stop	intermediateStop
noStop	No_stop	stop
resetError	Fault_acknowledgement	resetError
blockNumber	Block_number	blockNumber
veloOverride	Correction	velocityOverride
setStartInformation	Start_byte	setStartInformation
mdiMode	MDI_type	MDIMode
mdiVelocity	MDI_velocity	MDIVelocity
mdiAcceleration	MDI_acceleration	MDIAcceleration
mdiPosition	MDI_position	MDIPosition
reqControl	Ctrl_Req	-
pkwOut	I_O_address	PKWOutputInterface
pzdOut	I_O_address	PZDOutputInterface
busy	FB_coordination	busy
ready	Ready	ready
active	-	-

Nom utilisé dans le système SIMOTION V4.1 ou supérieure (bibliothèque d'instructions de SCOUT)	Nom dans le système SIMATIC	Nom dans le système SIMOTION jusqu'à la version V3.2 (SIMOTION Function Library)
dataReady	Data_transfer_ready	-
statusWord	Status_word	statusWord
actBlockNumber	Actual_block	actualBlockNumber
statusInformation	Checkback_signal_byte	statusInformation
driveWarning	Warning	driveWarning
driveWarnId	Warn_number	driveWarningNumber
driveWarnInfo	Warn_info	-
driveError	-	driveError
driveErrorId	-	driveErrorNumber
error	Fault	error
errorID	Fault_number	errorNumber
_POSMOA_nControl	FB 9	-
enable	-	-
pkwIn	-	-
pzdIn	-	-
init	Initialization	-
releaseBrake	Brake_release	-
accelTime	Acc_Time	-
releaseRamp	Ramp_en	-
startRamp	Ramp_on	-
setpoint	Sp	-
releaseSetpoint	Sp_en	-
resetError	Fault_acknowledgement	-
reqControl	Ctrl_req	-
busy	FB_coordination	-
pkwOut	-	-
pzdOut	-	-
ready	Ready	-
active	-	-
dataReady	Data_transfer_ready	-
statusWord	Status_word	-
actValue	Pv	-
error	-	-
errorID	-	-
driveError	Fault	-
driveErrorId	Fault_number	-
driveWarning	Warning	-
driveWarnId	Warn_number	-
driveWarnInfo	Warn_info	-

Nom utilisé dans le système SIMOTION V4.1 ou supérieure (bibliothèque d'instructions de SCOUT)	Nom dans le système SIMATIC	Nom dans le système SIMOTION jusqu'à la version V3.2 (SIMOTION Function Library)
_POSMA_rwParameter	FB 11	_FB_posmoA_readWriteParameter
pkwIn	I_O_address	PKWInputInterface
paraNumber	Number	parameterNumber
subindex	Index	subindex
enableRead	Read	read
execWrite	Write	write
setDefault	Factory_default	setFactorySettings
saveParameter	Parameter_save	saveParameter
pkwOut	I_O_address	PKWOutputInterface
busy	FB_coordination	busy
realValue	Value	REALValue
dintValue	Value	DINTValue
done	Task_completed	done
error	Fault_present	error
errorID	Fault_number	errorNumber
_POSMA_rwAllParameter	FB 12	_FB_posmoA_readWriteAllParameter
pkwIn	I_O_address	PKWInputInterface
execRead	Read_all	read
execWrite	Write_all	write
posMode	Pos_en	-
pkwOut	I_O_address	PKWOutputInterface
allParaPosmoA	-	allPosmoAParameter
busy	FB_active	busy
done	Task_complete	done
error	Fault_present	error
errorID	Fault_number	errorNumber
errorParaNumber	Fault_parameter_number	errorParameterNumber
Éléments des structures de données		
Struct_POSMA_params		Struct_posmoA_parameter
p1	p1	parameter1
p2	p2	parameter2
p3	p3	parameter3
p4	p4	parameter4
p5	p5	parameter5
p6	p6	parameter6
p7	p7	parameter7
p8	p8	parameter8
p9	p9	parameter9

Nom utilisé dans le système SIMOTION V4.1 ou supérieure (bibliothèque d'instructions de SCOUT)	Nom dans le système SIMATIC	Nom dans le système SIMOTION jusqu'à la version V3.2 (SIMOTION Function Library)
p10	p10	parameter10
p11	p11	parameter11
p12	p12	parameter12
p13	p13	parameter13
p14	p14	parameter14
p15	p15	parameter15
p16	p16	parameter16
p17	p17	parameter17
p18	p18	parameter18
p19	p19	parameter19
p20	p20	parameter20
p21	p21	parameter21
p22	p22	parameter22
p23	p23	parameter23
p24	p24	parameter24
p25	p25	parameter25
p26	p26	parameter26
p27	p27	parameter27
p28	p28	parameter28
p29	p29	parameter29
p30	p30	parameter30
p31	p31	parameter31
p32	p32	parameter32
p33	p33	parameter33
p34	p34	parameter34
p35	p35	parameter35
p36	p36	parameter36
p37	p37	parameter37
p38	p38	parameter38
p39	p39	parameter39
p40	p40	parameter40
p41	p41	parameter41
p42	p42	parameter42
p43	p43	parameter43
p44	p44	parameter44
p45	p45	parameter45
p46	p46	parameter46
p47	p47	parameter47
p48	p48	parameter48
p49	p49	parameter49
p50	p50	parameter50

Nom utilisé dans le système SIMOTION V4.1 ou supérieure (bibliothèque d'instructions de SCOUT)	Nom dans le système SIMATIC	Nom dans le système SIMOTION jusqu'à la version V3.2 (SIMOTION Function Library)
p51	p51	parameter51
p52	p52	parameter52
p53	p53	parameter53
p54	p54	parameter54
p55	p55	parameter55
p56	p56	parameter56
p57	p57	parameter57
p58	p58	parameter58
p59	p59	parameter59
p60	p60	parameter60
p61	p61	parameter61
p62	p62	parameter62
p80	p80	parameter80
p81	p81	parameter81
p82	p82	parameter82
p83	p83	parameter83
p84	p84	parameter84
p85	p85	parameter85
p86	p86	parameter86
p87	p87	parameter87
p99	p99	parameter99
p100	p100	parameter100
p101	p101	parameter101
p700	p700	-
p701	p701	-
p880	p880	-
p930	p930	parameter930
p964	p964	parameter964
p1426	p1426	-
p1427	p1427	-
Mot de commande de programme		
Array_POSMOA_prgCtrlInfo		Array_posmoA_programControlInformation

A.2 Liste des abréviations

Tableau A- 2 Abréviations

Abréviation	Signification
DC	Courant continu
Maître	Périphérie décentralisée
EEPROM	Electrically Erasable Programmable Read-Only Memory (mémoire morte programmable et effaçable électriquement)
ES	Système d'ingénierie
FB	Bloc fonctionnel
FW	Firmware
HW	Hardware (matériel)
IN	Paramètres d'entrée
IN/OUT	paramètre d'entrée/sortie
CONT	Schéma à contacts
MDI	M anual D ata I ntput (Introduction manuelle des données)
OUT	Paramètre de sortie
PKW	Paramètre, identificateur, valeur : Partie paramètres d'un PPO
POSMO A	P ositioning M otor A ctuator (moteur de positionnement)
PPO	P aramètres D onnées e P rocessus O bjet : télégramme de données cyclique en transmission sur PROFIBUS DP avec le profil "Entraînements à vitesse variable"
PZD	Données de processus Partie données process d'un PPO
ST	Structured Text
STW	Mot de commande
SW	Logiciel
TO	Objet technologique

Index

A

Appel des blocs fonctionnels
 _POSMOA, 42

B

Bibliographie, 3
Blocs fonctionnels POSMO A
 _POSMOA_control, 17
 _POSMOA_nControl, 24
 _POSMOA_rwAllParameter, 35
 _POSMOA_rwParameter, 31
Exemple d'appel, 42
Vue d'ensemble, 17

D

Descripteurs SIMOTION et SIMATIC, 55

E

Exemple d'application POSMO A, 45

I

Intégration des blocs fonctionnels, 14

M

Mise en service, 12

S

Structure de données Struct_POSMOA_params, 37
Substitution de télégramme
 FB_POSMOA_control, 23
 FB_POSMOA_nControl, 28

V

Variable E/S
 Création, 15