

Module de moteur pas à pas PCD2/3.H22

0	Table des matières	
0.1	Historique de document	0-3
0.2	Marques	0-3
1	Introduction	
1.1	Fonctionnement et utilisation	1-1
1.2	Les principales caractéristiques	1-2
1.3	Domaines d'application typiques	1-2
1.4	Dimensions	1-2
2	Caractéristiques techniques	
2.1	Caractéristiques techniques du matériel	2-1
2.1.1	Alimentation 24 VCC	2-1
2.1.2	Alimentation 5 VCC au bus PCD1/2/3	2-1
2.1.3	Entrées	2-1
2.1.4	Sorties	2-1
2.1.5	Connexions bidirectionnelles	2-2
2.1.6	Conditions de service	2-2
2.1.7	Données de commande	2-2
2.2	Bornes	2-3
2.3	Affichage à DEL	2-4
3	Description du fonctionnement	
3.1	Paramétrage d'un profil de course	3-1
3.1.1	Correction automatique de profil	3-2
3.2	Indication de la longueur de course	3-2
3.2.1	Indication de longueur de course relative	3-2
3.2.2	Indication de cible absolue	3-2
3.3	Comportement en cas d'arrêt d'urgence (Emergency Stop)	3-2
3.4	Comportement en cas d'actionnement d'un fdc en cours de marche	3-3
3.5	Démarrage d'un axe sur événement	3-3
3.6	Positionnement sur contacteur fin de course	3-3
3.7	Course de référence	3-4
3.7.1	Recherche du fin de course, suivie de la course jusqu'au contacteur de référence	3-4
3.7.2	Recherche du contacteur de référence au-delà des contacteurs de fin de course	3-4
3.7.3	Atteinte et dépassement du contacteur de référence	3-5
3.8	Incréments unitaires	3-5
3.9	Positionnement à vitesse constante	3-5
3.10	Positionnement sur profil de course	3-5
3.11	Arrêt d'un mouvement	3-5

4	Fonctions programmables dans PG5	
4.1	Accès par des blocs de fonctions (FB)	4-1
4.1.1	Indicateur et registre de diagnostic	4-2
4.1.2	FB init.....	4-2
4.1.3	FB initMP	4-3
4.1.3	FB exec.....	4-4
4.1.5	Commandes.....	4-6
4.1.6	Instructions d'écriture.....	4-7
4.1.7	Instructions de lecture.....	4-8
4.2	Configuration du système avec le Configurateur d'appareil	4-11
4.3	Propriétés.....	4-13
4.4	Jeu d'instructions - Accès direct.....	4-14
4.4.1	Instructions d'écriture.....	4-14
4.4.2	Instructions de lecture.....	4-16
5	Adaptations du programme en cas de remplacement d'un module H210 par un module H222	
6	Exemples	
6.1	Matériel PCD2	6-1
6.2	Matériel PCD3	6-2
6.3	Description du programme.....	6-3
6.4	Code du programme IL	6-3
A	Annexe	
A.1	Icones	A-1
A.2	Adresse de la société Saia-Burgess Controls AG.....	A-2

0.1 Historique de document

0

Version	Traitement	Publication	Remarques
FR01	2011-01-03	2011-04-15	Première édition
FR02	2011-08-25	2011-08-26	- Nouveau numéro de Téléphone: +41 26 672 72 72 Corrigé le nom de la variable : « IO.Slot0.IOAccess.X_AXIS_RD_ACTPOSABS »
FR03	2013-06-17	2013-08-14	Changement du logo et nom
FR04	2014-02-19	2014-02-19	Plusieurs changements dans le manuel entier
FR05	2017-05-30	2017-05-30	- Nouveau numéro de Téléphone: +41 26 580 30 00 - Ligne « IO.Slot0.IOAccess.X_AXIS_RD_WARN » de 4.4.2 supprimé.

0.2 Marques

Saia PCD® est une marque déposée de Saia-Burgess Controls AG.

Sous réserve d'adaptations et de modifications techniques selon l'état de la technique.

Saia-Burgess Controls AG, 2017. © Tous droits réservés.

Publié en Suisse

1 Introduction

Ce module est enfichable dans tout emplacement d'E/S d'une PCD2 ou PCD3. Il permet de commander le niveau de puissance d'un axe de moteur pas à pas jusqu'à une fréquence de 20 kHz.



Le nombre maximum de modules est déterminé par la limite des extensions PCD2 ou PCD3. Aucun module ne doit occuper l'adresse 255.

1.1 Fonctionnement et utilisation

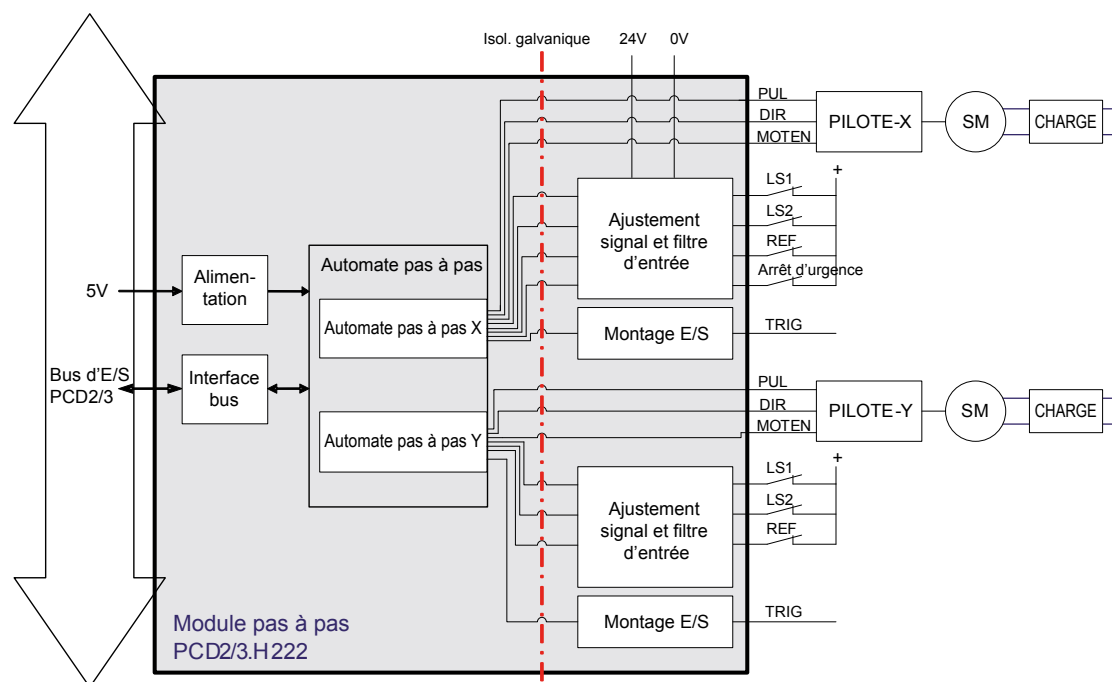
Le module PCD2/3.H222 permet de commander deux étages de sortie de moteur pas à pas. Il prend en charge la commande des mouvements souhaités et transmet à l'étage de sortie les trois signaux MOTEN (activation), DIR (sens de rotation) et PUL (impulsion). Outre les incréments unitaires et plusieurs incréments à vitesse constante, il est possible d'exécuter des profils de déplacement trapézoïdaux ou en S à rampes d'accélération et de freinage asymétriques. Les positionnements absolus ou relatifs sont possibles.

Chaque axe dispose de trois entrées pour LS1 (contacteur de fin de course), LS2 (contacteur de fin de course) et REF (contacteur de référence). Ceci permet de déclencher le positionnement par une seule instruction envoyée au contacteur correspondant.

Un connecteur TRIG (« trigger » ou déclencheur) configurable par axe permet de déclencher l'exécution d'un mouvement par un signal 24 V quelconque. Lorsque le connecteur TRIG est utilisé comme entrée pour un axe et comme sortie pour un autre, les profils de mouvement peuvent démarrer de manière synchrone en agissant comme une commande maître par rapport aux modules.

Chaque module dispose d'une autre entrée pour la détection des arrêts d'urgence. Toutes les entrées et sorties sont isolées galvaniquement du bus d'E/S.

Diagramme synoptique d'une entraînement par moteur pas à pas



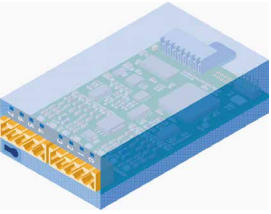
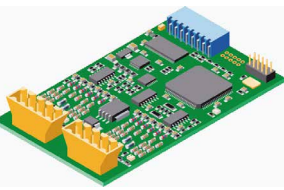
1.2 Les principales caractéristiques

- Une entrée d'arrêt d'urgence commune
- 3 entrées par axe (1 contacteur de référence et 2 contacteurs de fin de course)
- 3 sorties par axe (Pulse, Dir, MotEn)
- Ein konfigurierbarer Synchronisier-Ein-/Ausgang pro Achse
- Possibilités de positionnement : incrément unitaire, vitesse constante, profils de mouvement ; automatisch auf Referenz- oder Endschalter fahren
- 16 profils par axe pour le paramétrage d'une courbe en S ou trapézoïdale metrischer Start- und Bremsrampe
- Indication de la position relative ou absolue.
- Mouvements continus.

1.3 Domaines d'application typiques

- Handling- und Montageautomaten
- Fonctions de prise et de pose
- Entraînements économiques de palletisation et de montage.
- Commande angulaire automatique (caméras, projecteurs, antennes, etc.)
- Positionnement d'axes statiques (set up).

1.4 Dimensions

<p>Module PCD3</p> 	<p>Dimension : module d'E/S PCD3</p> <p>Couleur du boîtier : bleu</p> <p>Raccordement : 2 borniers à ressort 10 broches enfichables</p>
<p>Module PCD2</p> 	<p>Dimension : module d'E/S PCD3</p> <p>Raccordement : 2 borniers à ressort 10 contacts enfichables</p>

2 Caractéristiques techniques

2.1 Caractéristiques techniques du matériel

2.1.1 Alimentation 24 VCC

Paramètre	Valeur
Plage de tension	18...32 VCC
Consommation de courant à 24 V	< 80 mA
Protection anti-surtension	Oui (39 V \pm 10 %)
Protection en cas d'inversion de phase	Non

2.1.2 Alimentation 5 VCC au bus PCD1/2/3

Paramètre	Valeur
Consommation de courant à 5V	< 85 mA

2.1.3 Entrées

Paramètre	Valeur
Total	7
Plage de tension	0 ... 32 V
Plage Bas	0 ... 5 V
Plage Haut	15 ... 32 V
Seuil de commutation Bas-Haut	env. 10,9 V
Seuil de commutation Haut-Bas	env. 9,1 V
Hystérésis	env. 1,8 V
Courant d'entrée (24 V)	env. 5,2 mA
Retard à l'enclenchement	env. 200 μ s
Retard au déclenchement	env. 200 μ s

2.1.4 Sorties

Paramètre	Valeur
Total	6
Impulsions unitaires (PUL)	Actif haut
Durée d'impulsion du signal unitaire	16,5 μ s
Signal de direction (DIR)	Bas = avance (sens horaire [cw]) Haut = recul (sens antihoraire [ccw])
Signal de mouvement (MOTEN)	Haut = moteur activé Bas = moteur désactivé
Charge par sortie	< 20 mA
Mode de fonctionnement	Récepteur (commutation sur moins)
Retard à l'enclenchement MOTEN, DIR	< 15 μ s
Retard au déclenchement MOTEN, DIR	< 25 μ s
Retard à l'enclenchement PUL	< 1,5 μ s
Retard au déclenchement PUL	< 500 ns

2.1.5 Connexions bidirectionnelles

Il existe deux connexions bidirectionnelles :

entrée/sortie TRIG axe X
entrée/sortie TRIG axe Y.

Ces connexions sont utilisables en entrée ou en sortie. Ceci permet de déterminer un axe maître pour la synchronisation des axes esclaves.

Caractéristiques d'utilisation en entrée

Paramètre	Valeur
Plage de tension	0 ... 32 V
Plage Bas	0 ... 5 V
Plage Haut	15 ... 32 V
Seuil de commutation Bas-Haut	env. 9,3 V
Seuil de commutation Haut-Bas	env. 11 V
Hystérésis	env. 1,7 V
Courant d'entrée (24 V)	3,85 mA
Retard à l'enclenchement	< 170 µs
Retard au déclenchement	< 184 µs

Caractéristiques d'utilisation en sortie

Paramètre	Valeur
Plage de tension	0 ... 32 V
Plage Bas	0 ... 5 V
Plage Haut	15 ... 32 V
Charge par sortie	> 200 mA
Retard à l'enclenchement	< 1,5 µs
Retard au déclenchement	< 28 µs
Mode de fonctionnement	Générateur

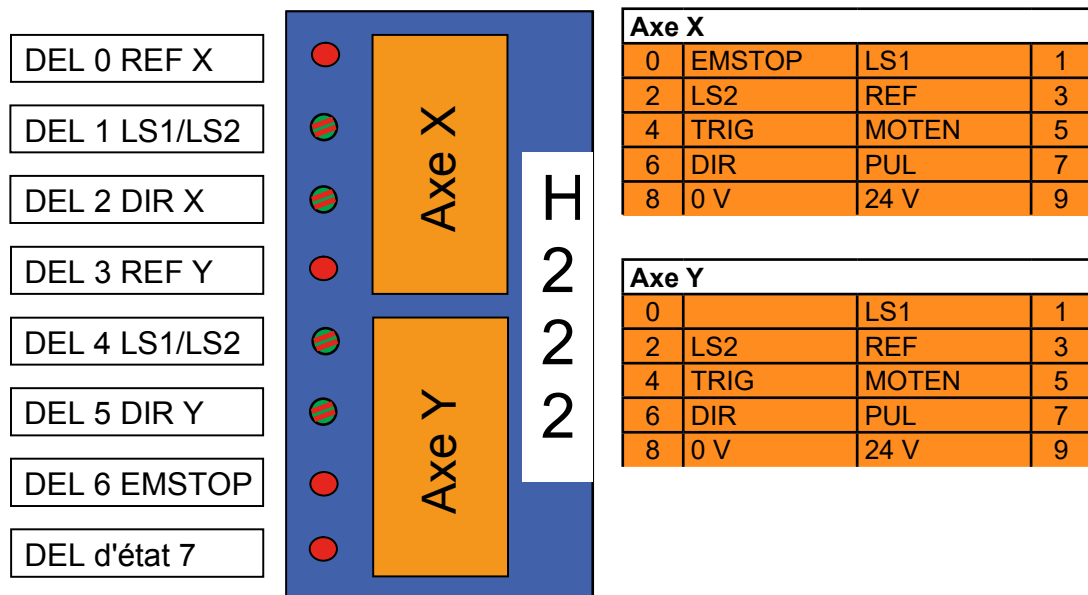
2.1.6 Conditions de service

température ambiante	Service : 0...+55 °C sans ventilation forcée. Entreposage -20...+85 °C
----------------------	--

2.1.7 Données de commande

Type	Description	Poids
PCD2.H222	Module pas à pas pour 2 axes	27 g
PCD3.H222	Module pas à pas pour 2 axes	70 g

2.2 Bornes



Un contact d'arrêt d'urgence peut être raccordé à l'entrée EMSTOP. La figure indique l'affectation des bornes. Cette entrée permet de détecter une déconnexion pour arrêt d'urgence et ne doit pas être utilisée pour une fonction de sûreté. Les connexions LS1, LS2 et REF sont des entrées prévues pour le positionnement sur un des contacteurs. En outre, LS1 et LS2 limitent la plage utile d'un axe de moteur pas à pas. Ces entrées peuvent être configurées pour utiliser aussi bien un contact travail qu'un contact repos et pour désactiver un contacteur non raccordé, d'où une meilleure détection des erreurs.

Lorsque l'une de ces entrées est désactivée, elle peut être utilisée comme entrée numérique. A cet effet, le registre d'état des E/S doit être lu par le programme utilisateur.

La connexion TRIG est configurable en entrée ou en sortie et sert à démarrer un axe sur événement.

Lorsque la connexion est activée et configurée en entrée, le mouvement démarré par le programme utilisateur ne démarre que lorsque 24 V sont appliqués à la connexion TRIG.

Lorsqu'elle est utilisée en sortie, la connexion TRIG est connectée et déconnectée en même temps que le signal de sortie MOTEN.

MOTEN, DIR et PUL sont des sorties utilisées en mode consommateur pour commander l'étagé de sortie du moteur pas à pas.

Le module est alimenté en courant par la connexion +24 V et 0 V. Ces deux signaux sont reliés entre eux en interne par les bornes des axes X et Y.

Sauf pour les signaux MSTOP, +24 V et 0 V, tous les signaux sont présents une fois pour l'axe X et une fois pour l'axe Y.

2.3 Affichage à DEL

Axe X

DEL 0	Tension à l'entrée REF
DEL 1 (rouge)	Tension à l'entrée LS1
DEL 1 (vert)	Tension à l'entrée LS2
DEL 2 (rouge)	Tension aux sorties MOTEN et DIR
DEL 2 (vert)	Tension à la sortie MOTEN et 0 V à la sortie DIR

Axe Y

DEL 3	Tension à l'entrée REF
DEL 4 (rouge)	Tension à l'entrée LS1
DEL 4 (vert)	Tension à l'entrée LS2
DEL 5 (rouge)	Tension aux sorties MOTEN et DIR
DEL 5 (vert)	Tension à la sortie MOTEN et 0 V à la sortie DIR

Pour les deux axes

DEL 6 :	Tension à l'entrée EMSTOP
DEL 7 :	Visualisation des états de fonctionnement et des codes d'erreur

EMSTOP et REF sont visualisées par une DEL monocolore. LS1 et LS2 sont signalées par une DEL bicolore (rouge pour LS1 et verte pour LS2). L'état de DIR et de MOTEN est également signalé par une DEL bicolore. Si MOTEN = 1, la couleur de la DEL dépend du sens de rotation DIR (rouge pour le sens antihoraire « ccw » et vert pour le sens horaire « cw »).

La DEL 7 signale une erreur éventuelle sur l'un des deux axes.

3 Description du fonctionnement

Le module H222 s'utilise avec une PCD2 ou une PCD3 et est commandé par un accès périphérique à partir du programme utilisateur. Cet accès déclenche la communication entre le PCD et le H222 par le bus d'E/S standard. Le module détecte automatiquement les nouvelles instructions de course et engage les opérations nécessaires pour leur traitement.

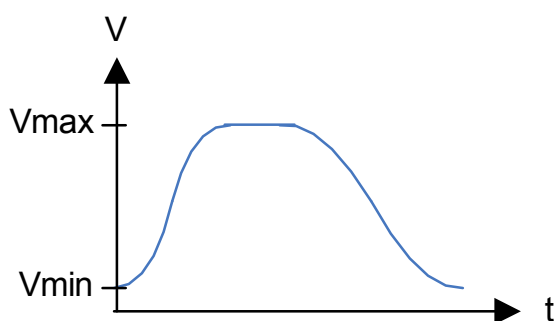
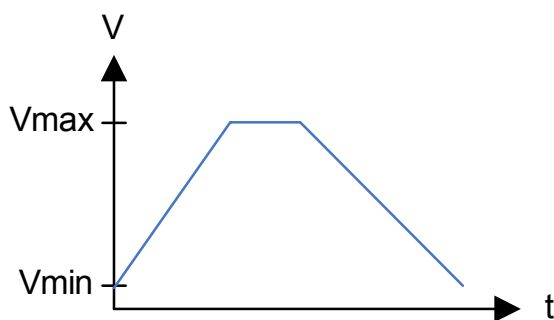
3.1 Paramétrage d'un profil de course

Afin d'éviter au PCD de devoir envoyer au module à chaque fois tous les paramètres d'un profil de course, 16 profils peuvent être définis et mémorisés une fois pour toutes.

Un profil de course est défini par les paramètres suivants :

Paramètre	Description	Plage, unité	Résolution, remarque
Vmin	Vitesse de démarrage/arrêt	10...10 000 Hz	1 Hz (16 bits)
Vmax	Vitesse maximale	20...20 000 Hz	1 Hz (16 bits)
Acc	Accélération moyenne	1...1000 kHz/s	1 kHz (16 bits)
Dec	Décélération moyenne	1...1000 kHz/s	1 kHz (16 bits)
Jerk_acc	Pourcentage de jerk de la rampe de démarrage	0...50 %	1 % (6 bits)
Jerk_dec	Pourcentage de jerk de la rampe de décélération	0...50 %	1 % (6 bits)

Les deux valeurs de jerk indiquent la durée en pourcentage de la phase d'accélération ou de freinage à l'issue de laquelle l'accélération ou la décélération linéaires doivent être atteintes. La durée de la phase linéaire est alors égale à $100\% - 2 \times \text{valeur de jerk}$. Une valeur de jerk de 0 % correspond à un profil de course trapézoïdal, tandis qu'une valeur de 50 % correspond à une courbe en S.



La durée d'exécution de chacun des profils est déterminée par ces paramètres et par la longueur de course.

3.1.1 Correction automatique de profil

Si des valeurs de réglage se situent hors de la plage admissible, le module les corrige automatiquement en les remplaçant par la valeur minimale ou maximale selon le cas. Un signal d'avertissement est émis. Si la valeur de V_{min} , V_{max} , Acc ou Dec est égale à 0, le profil n'est pas exécuté et un message d'erreur est émis.

Il existe entre certains paramètres d'entrée des relations qui doivent être respectées. Dans le cas contraire, il se peut, par exemple, qu'une courbe en S réglée soit ramenée à un profil trapézoïdal pour éviter des erreurs d'indexation.

3.2 Indication de la longueur de course

La longueur de course est indiquée indépendamment du profil de course, qu'il soit relatif, absolu ou continu, ce dernier étant un cas spécial d'indication d'une longueur de course relative. Les indications de longueur de course influent sur les modes de positionnement « à vitesse constante » et « sur profil de course ».

3.2.1 Indication de longueur de course relative

Une longueur de course relative est indiquée en nombre d'incrémentes à exécuter. Le sens de rotation est alors à indiquer au moment du démarrage du mouvement. Le nombre admissible d'incrémentes est situé entre 0 et 16 777 215, ce dernier nombre correspondant à un mouvement continu. Toutes les autres valeurs induisent le nombre d'incrémentes correspondant.

3.2.2 Indication de cible absolue

Dans un positionnement absolu, le point final peut être indiqué sans que la position réelle soit connue. Le module calcule lui-même la longueur de course relative et le sens de rotation en fonction de la position absolue. Au démarrage du mouvement, le paramètre de transfert « sens de rotation » est négligé.

3.3 Comportement en cas d'arrêt d'urgence (Emergency Stop)

L'entrée d'arrêt d'urgence permet d'arrêter un mouvement en cours avec une rampe de freinage prédéfinie sur les deux axes. Etant donné que la position réelle ne correspond plus après une mise hors circuit rapide déclenchée en externe, l'un des contacteurs de fin de course ou le contacteur de référence doit être actionné après un arrêt d'urgence.



Une mise hors circuit rapide liée à une fonction de sûreté doit intervenir indépendamment du module.

3.4 Comportement en cas d'actionnement d'un fdc en cours de marche

L'actionnement de LS1 ou LS2 déclenche automatiquement une rampe de freinage. La position absolue actuelle n'est pas perdue. Un message d'erreur est toutefois émis, lequel doit être acquitté. Un contacteur de fin de course (fdc) ne peut quitter sa plage d'actionnement que moyennant un positionnement sur lui ou par l'exécution de la course de référence.

3.5 Démarrage d'un axe sur événement

Lorsque la connexion TRIG est configurée en entrée et activée, les instructions de course peuvent être lancées pour un axe dont le démarrage sera conditionné par l'application de 24 V à la connexion TRIG. N'importe quel signal numérique de 24 V peut être utilisé à cet effet.

Si deux ou plusieurs axes doivent démarrer en synchronisme, la connexion TRIG doit être configurée en entrée et activée pour les axes consécutifs. Pour l'axe maître, la connexion est activée comme sortie. Toutes les connexions doivent être reliées ensemble par un câble en externe. Les consignes de course de tous les axes consécutifs sont lancées d'abord. Les axes attendent ensuite l'apparition du signal de démarrage à la connexion TRIG, lequel intervient lorsque l'axe maître démarre à l'instant correspondant.

3.6 Positionnement sur contacteur fin de course

Dans le cas d'une course sur contacteur fin de course, deux positions initiales sont à distinguer : ou bien il s'agit d'une course qui se déroule entre deux fins de course, en direction de l'un d'entre eux avec le profil de vitesse réglé, avec freinage jusqu'à l'arrêt dès l'actionnement du fin de course et redémarrage avec V_{min} dans le sens opposé jusqu'à relâchement du fin de course, ou bien il s'agit d'une course sur la plage active de l'indicateur du fin de course. Dans ce cas, la course s'effectue directement à V_{min} sur la plage valide entre les fins de course jusqu'à relâchement du fin de course.

Il n'est pas possible d'actionner directement le fin de course le plus proche hors de la plage valide (au-delà des deux fins de course). Pour ce faire, la plage valide doit d'abord être parcourue.

Le positionnement sur l'un des deux fins de course est lancé par l'accès périphérique et, selon l'axe souhaité, par l'instruction X_AXIS_WR_LSREFMOVE ou Y_AXIS_WR_LSREFMOVE. Le fin de course sur lequel doit s'effectuer le positionnement est déterminé par le paramètre de transfert (1 = LS1, 2 = LS2).

3.7 Course de référence

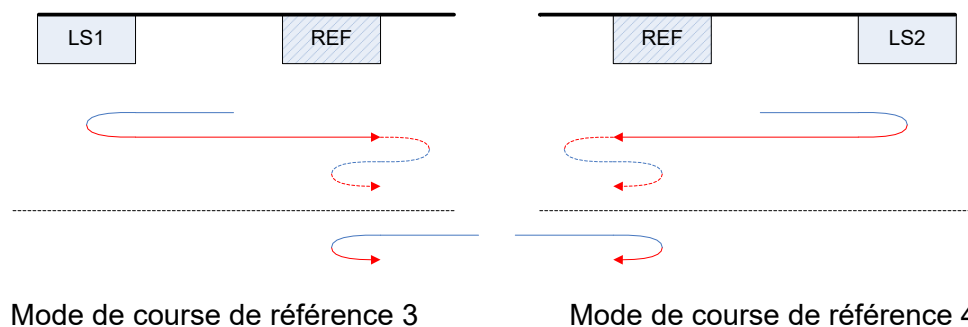
Il existe plusieurs modes de positionnement sur le contacteur de référence connecté. Ce dernier doit être actionné quel que soit le mode ; un message d'erreur est émis dans le cas contraire.

3.7.1 Recherche du fin de course, suivie de la course jusqu'au contacteur de référence

3

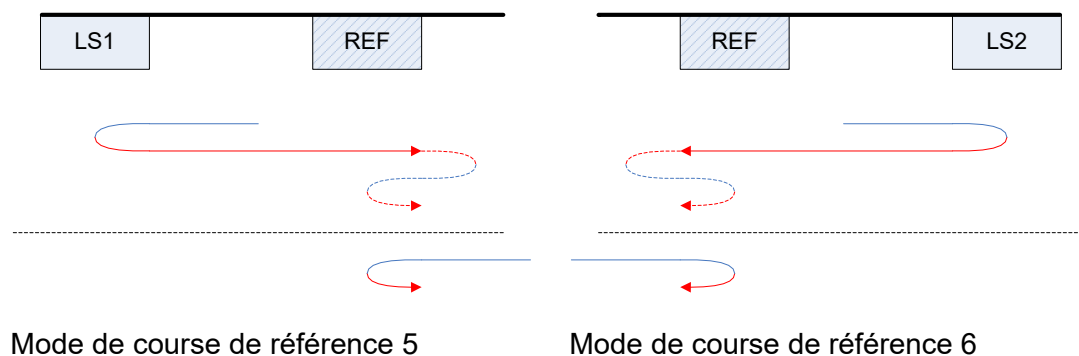
Si la référence est prise dans le sens d'avance (mode de course de référence 3), la course s'effectue d'abord selon le profil de vitesse actif jusqu'à l'actionnement du contacteur de référence ou du contacteur de fin de course, ce qui déclenche le freinage. Une fois à l'arrêt, le sens de déplacement est inversé et la course s'effectue selon le profil de vitesse jusqu'au contacteur de référence. La décélération commence à l'actionnement du contacteur de référence. Si c'est l'autre contacteur qui est actionné, la course de référence s'interrompt et un défaut est affiché.

Si le contacteur de référence est actionné à l'issue de la rampe de décélération, la course se poursuit à V_{min} sans interruption jusqu'au relâchement du contacteur de référence. Dans le cas contraire, la course s'effectue d'abord à V_{min} dans le sens inverse jusqu'au contacteur de référence et, seulement ensuite, dans le sens d'avance pour quitter la plage d'actionnement du contacteur.



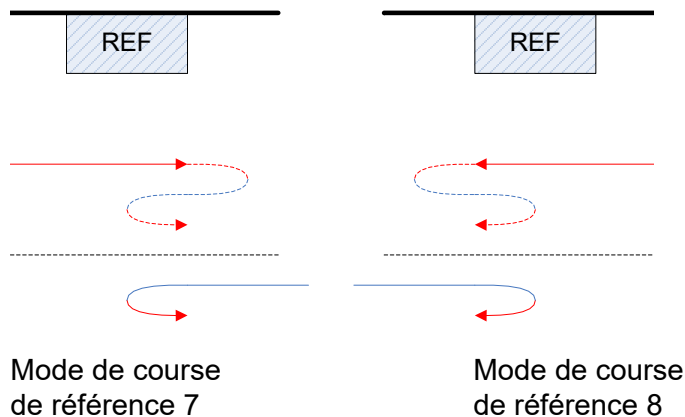
3.7.2 Recherche du contacteur de référence au-delà des contacteurs de fin de course

Le mode de course de référence 5 permet de monter le contacteur de référence au-delà des fins de course et d'aller jusqu'au contacteur de référence bien que les fins de course soient activés.



3.7.3 Atteinte et dépassement du contacteur de référence

Dans ce mode de référencement, le démarrage s'effectue directement dans le sens de rotation final, avec dépassement du contacteur de référence. Si, en mode de course de référence 7, LS1 est actionné (dans la mesure où il est activé), la course de référence s'interrompt et un message d'erreur est émis. Il en va de même pour le mode 8 et LS2.



3.8 Incréments unitaires

Tout incrément unitaire nécessite un ordre du PCD. Celui-ci est émis avec la valeur de vitesse définie pour V_{min} . En l'absence de ce réglage, l'incrément unitaire est fixé à 10 Hz. La cadence incrémentielle maximale admissible en mode pas à pas est égale à 350 Hz.

3.9 Positionnement à vitesse constante

Un mouvement qui démarre à V_{min} utilise le paramètre V_{min} et le nombre d'incrément à exécuter. Si ce nombre est fixé à 16 777 215 ($2^{24}-1$), le mouvement démarre à V_{min} et ne s'arrête qu'à l'atteinte du contacteur fin de course correspondant ou à la réception d'une instruction d'arrêt émis par le programme utilisateur.

3.10 Positionnement sur profil de course

Ce mode de positionnement, qui est le plus courant, est celui qui permet d'atteindre le plus rapidement la position cible. Il utilise les paramètres de profil de course actuels et un profil en courbe en S, trapézoïdal, triangulaire ou à vitesse constante est exécuté en fonction du réglage des paramètres et de la distance de positionnement.

3.11 Arrêt d'un mouvement

Un mouvement lancé peut être interrompu par une instruction d'arrêt. L'axe est mis à l'arrêt selon la rampe de freinage prédéfinie. Cette instruction déclenche un message d'erreur qui doit être acquitté avant la prochaine instruction de course.

4 Fonctions programmables dans PG5



Soyez attentif aux différentes possibilités de commande du module par accès direct (Media Mapping) ou blocs de fonction (FB).

4.1 Accès par des blocs de fonctions (FB)

Le module H222 est intégré dans un PCD2 ou un PCD3 et commandé à l'aide de blocs de fonction (FB) dans l'outil de programmation PG5. Ces FB déclenchent une communication entre le PCD et le module H222 via le bus d'E/S standard. Ainsi, les paramètres de déplacement et les instructions de course sont définis dans le registre correspondant du microcontrôleur sur le module d'extension.

Le microprogramme du module détecte automatiquement les nouvelles instructions de course et engage les mesures nécessaires à leur traitement. Afin d'éviter au SPS de devoir envoyer à chaque fois au module tous les paramètres d'un profil de course, 16 profils activés par une commande unique peuvent être définis et mémorisés une fois pour toutes.

Le module permet de raccorder des contacteurs d'arrêt d'urgence, de fin de course et de référence. Une mise hors circuit rapide liée à une fonction de sûreté doit toutefois intervenir de manière externe. Ces entrées peuvent être configurées pour utiliser aussi bien un contact travail qu'un contact repos et pour désactiver un contacteur non raccordé, d'où une sécurité accrue grâce à une meilleure détection des erreurs.

Le module est initialisé à l'aide d'un bloc de fonction « Init ». Le bloc de fonction « InitMP » permet de transmettre un profil de course complet avec un appel de FB. Le FB « Exec » permet de transmettre une commande (« command »). En fonction du type d'instruction, un octet de données est fourni ou lu en tant que paramètre de retour. Une liste de toutes les commandes admissibles et leur description figure dans le chapitre

4.1.1 Indicateur et registre de diagnostic

Le registre de diagnostic et l'indicateur de diagnostic sont mis à jour pour chaque axe initialisé après les actions correspondantes.

Structure du registre de diagnostic rDiag :

```

;          31.....24 23.....16 15.....8 7.....0
;          \ Error / \ FB No. / \ Par.No. / \Module No./
;
;          Error:      00h -> No special error
;                      01h -> H222 FB already in use
;          FB No.:    01h -> H222.Init
;                      02h -> H222.InitMP
;                      03h -> H222.Exec
;          Par.No.:   00h -> No parameter error
;                      01h -> Module number not correct
;                      02h -> Axis number not correct
;                      03h -> Command code not correct (<0 or >255)
;          Module No: Number of the Module (FB parameter 1) which
;                      causes the error

```

4

4.1.2 FB init

Ce FB initialise tous les modules H222 qui ont été définis dans le fichier .equ correspondant. fDiag indique si une erreur s'est produite ou pas pendant l'initialisation. rDiag permet de l'identifier.

L'appel du FB « INIT » se présente comme suit :

```

CFB  H222.Init
R  rDiag      ; Par. 1 : registre de diagnostic
F  fDiag      ; Par. 2 : indicateur d'état (0 : Init OK, 1 :
erreur)

```

Ou représenté sous forme d'appel de FB :



4.1.3 FB initMP

Ce FB initialise un profil de déplacement (MP : motion profile) et configure les contacteurs d'arrêt d'urgence, de fin de course ou de référence. rDiag fournit un acquittement qui permet de déterminer la cause en cas d'erreur. fDiag indique si une erreur s'est produite ou pas pendant l'initialisation.

L'appel du FB « INITMP » se présente comme suit :

CFB H222.InitMP

R modNr ; Par. 1 : adresse de base du module

R axisNr ; Par. 2 : numéro de l'axe (0 = axe X, 1 = axe Y)

R Vmin ; Par. 3 : Vmin en Hz

R Vmax ; Par. 4 : Vmax en Hz

R Accel ; Par. 5 : accélération en kHz/s

R Decel ; Par. 6 : décélération en kHz/s

R JerkAcc ; Par. 7 : pourcentage de jerk de l'accélération (0 à 50 %)

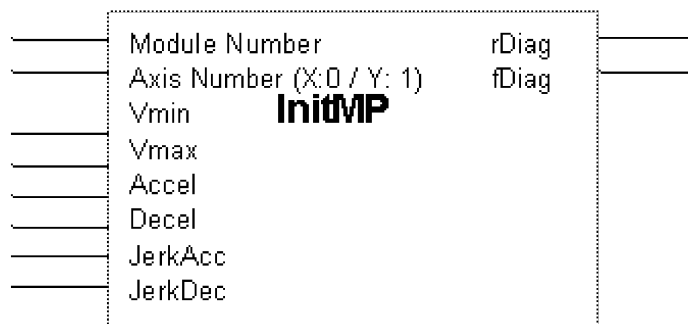
R JerkDec ; Par. 8 : pourcentage de jerk de la décélération (0 à 50 %)

R rDiag ; Par. 9 : registre de diagnostic

F fDiag ; Par.10 : indicateur d'état (0 : InitMP OK, 1 : erreur)

4

Ou représenté sous forme d'appel de FB :



4.1.3 FB exec

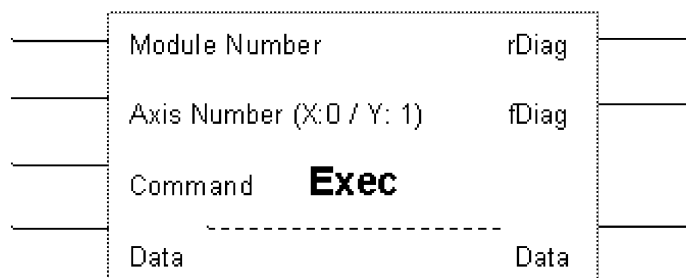
Le bloc individuel « exec » permet d'exécuter les instructions suivantes et de fournir d'autres données, comme le sens de rotation, etc. D'autres commandes fournissent des valeurs de retour (voir description des commandes). fDiag signale qu'une erreur s'est produite et rDiag permet de lui attribuer une cause.

L'appel du FB « EXEC » se présente comme suit :

```
CFB    H222.Exec
R modNr    ; Par. 1 : adresse de base du module
R axisNr   ; Par. 2 : numéro de l'axe (0 = axe X, 1 = axe Y)
R rCommand ; Par. 3 : commande (voir liste de commandes ci-dessous)
R rData    ; Par. 4 : paramètre de transfert ou lecture (en partie
             facultatif)
R rDiag    ; Par. 5 : registre de diagnostic
F fDiag    ; Par. 6 : indicateur d'état (0 : Exec OK, 1 : erreur)
```

4

Ou représenté sous forme d'appel de FB :



Le bloc FB exec prend en charge les instructions suivantes (Command) :

Commandes :

- Incrément unitaire (SingleStep)
- Course de référence ou rechercher et quitter LS ou quitter la plage Overrun (Ls-RefMove)
- Positionnement avec vitesse constante (MoveVconst)
- Positionnement avec profil de vitesse (StartMP)
- Annuler l'instruction de course (Stop)
- Validation des instructions de course après « Stop » (Cont)
- Sauvegarder le profil de course (SaveMP)
- Charger le profil de course (ActMP)
- Redémarrer le microprogramme du module (ResMC)
- Supprimer le registre d'erreur (ClrError)

Instructions de paramétrage :

- Entrer l'accélération moyenne au démarrage (LdAcc)
- Entrer la décélération de freinage moyenne (LdDec)
- Entrer la vitesse maximale (LdVmax)
- Entrer la vitesse minimale (LdVmin)
- Entrer la valeur de jerk pour la plage d'accélération (LdJPa)
- Entrer la valeur de jerk pour la plage de freinage (LdJPd)
- Entrer la position de consigne relative (LdDestRel)
- Entrer la position de consigne absolue (LdDestAbs)
- Définir la position absolue de l'emplacement actuel de l'axe (LdActPosAbs)
- Entrer la logique du contacteur de fin de course (LdIoConf)

4

Instructions de lecture :

- Lire l'accélération moyenne au démarrage (RdAcc)
- Lire la décélération de freinage moyenne (RdDec)
- Lire la vitesse maximale (RdVmax)
- Lire la vitesse minimale (RdVmin)
- Lire la valeur de jerk pour la plage d'accélération (RdJPa)
- Lire la valeur de jerk pour la plage de freinage (RdJPd)
- Lire la position de consigne relative (RdDestRel)
- Lire la position de consigne absolue (RdDestAbs)
- Lire la position réelle actuelle relative (RdActPos)
- Lire le temps de positionnement actuel (RdActRuntime)
- Lire la vitesse actuelle (RdActSpeed)
- Lire la position absolue actuelle (RdActPosAbs)
- Lire l'erreur (RdError)
- Lire l'état (RdStatus)
- Lire la configuration d'E/S (RdIoConf)
- Lire la version du microprogramme (RdFwVersion)

4.1.5 Commandes

Par.3 (rCommand)	Par.4 (rData)
StartMP	initialize and start motion Par.4 = 0: start actual motion profile cw 1: start actual motion profile ccw (endless move if LdDestRel with value 0xFFFFFFFF (16777215) was set)
Stop	stops motion (this instruction sets a HALT error in error register which needs to be cleared before next move instruction by Cont or ClrError instruction)
Cont	Clears HALT error in error register, which was set when Stop instruction was proceeded
SingleStep	single step with actual Vmin Par.4 . = 0: direction cw 1: direction ccw
MoveVconst	Positioning with min. frequency Par.4 . = 0: direction cw 1: direction ccw (endless move if LdDestRel with value 0xFFFFFFFF (16777215) was set)
LsRefMove	positioning on limit switch or reference switch border Par.4 . = 1: LS1 2: LS2 3: REF cw (search LS1 or REF in dir ccw) 4: REF ccw (search LS2 or REF in dir cw) 5: REF cw in invalid area under LS1(ccw first) 6: REF ccw in invalid area over LS2 (cw first) 7: REF cw (direct to REF) 8: REF ccw (direct to REF)
SaveMP	Save motion profile x (max. 16) Par.4 . = 0: profile 0 1: profile 1 .. 15: profile 15
ActMP	activate motion profile x (max. 16) Par.4 . = 0: profile 0 1: profile 1 .. 15: profile 15
ResMC	restart the module firmware, all profiles are deleted, errors are cleared (Par.2 and Par.4 have no impact)
ClrError	Clears error register

4.1.6 Instructions d'écriture

Par.3 (rCommand)	Par.4 (rData)
LdVmin	load min. frequency in Hz Par.4 = value (2 Bytes) (range: 10...10 000)
LdVmax	load max. frequency in Hz Par.4 = value (2 Bytes) (range: 20...20 000)
LdAcc	load acceleration in kHz/s Par.4 = value (2 Bytes) (range: 1...1 000)
LdDec	load deceleration in kHz/s Par.4 = value (2 Bytes) (range: 1...1 000)
LdJPa	load jerk value in percent for accel. Par.4 = value (2 Bytes) (range: 0...50)
LdJPd	load jerk value in percent for decel. Par.4 = value (2 Bytes) (range: 0...50)
LdIoConf	load input/output configuration Bit 0: Input LS1 on/off (0 = on/1 = off) Bit 1: Input LS2 on/off Bit 2: Input REF on/off Bit 3: Input Notstop on/off Bit 4: Input LS1 Mode (0=opening/1=closing cont.) Bit 5: Input LS2 Mode Bit 6: Input REF Mode Bit 7: Input Notstop Bit 8: use TRIG as input (0)/output (1) Bit 9: use TRIG (0 = TRIG not used/1 = TRIG used)
LdDestRel	load destination relative in steps Par.4 = value (3 Bytes) (range: 0...16 777 216)
LdDestAbs	load destination as absolute position Par.4 = value (4 Byte signed) (range: -2 147 483 648...+2 147 483 647 but take care that relative destination is smaller than 16 777 216!) LdDestAbs must be followed by MoveVconst or StartMP. Stop, SingleStep, LsRefMove will cancel LdDestAbs instruction.
LdActAbsPos	load absolute position for current motor position Par.4 = value (4 Byte signed) (range: -2 147 483 648...+2 147 483 647)

4.1.7 Instructions de lecture

Par.3 (rCommand)	Par.4 (rData)
RdVmin	read min. frequency in Hz Par.4 = value (2 Bytes)
RdVmax	read max. frequency in Hz Par.4 = value (2 Bytes)
RdAcc	read acceleration in kHz/s Par.4 = value (2 Bytes)
RdDec	read deceleration in kHz/s Par.4 = value (2 Bytes)
RdJPa	read jerk value in percent for accel. Par.4 = value (2 Bytes)
RdJPd	read jerk value in percent for decel. Par.4 = value (2 Bytes)
RdIoConf	read input/output configuration Par.4 = value (2 Bytes) Bit 0: Input LS1 on/off (0 = on/1 = off) Bit 1: Input LS2 on/off Bit 2: Input REF on/off Bit 3: Input Notstop on/off Bit 4: Input LS1 Mode (0=opening/1=closing cont.) Bit 5: Input LS2 Mode Bit 6: Input REF Mode Bit 7: Input Notstop Mode Bit 8: use TRIG as input (0)/output (1) Bit 9: use TRIG (0 = TRIG not used/1 = TRIG used)
RdStatus	read module status Bit 0: state of input LS1 Bit 1: state of input LS2 Bit 2: state of input REF Bit 3: state of input Notstop Bit 4: state of input/output TRIG Bit 5: state of output DIR Bit 6: state of output MOTEN Bit 7: - Bit 8: Wait for TRIG flag Bit 9: on destination flag
RdDestRel	read destination relative in steps actual used for next motion commands Par.4 = value (3 Bytes)
RdDestAbs	read destination absolute in steps actual used for next motion commands Par.4 = value (3 Bytes)
RdActSpeed	read actual speed in Hz Par.4 = value (2 Bytes)
RdActPos	read actual position in steps done since start of motion Par.4 = value (3 Bytes)
RdActPosAbs	Read actual absolute position Par.4 = value (4 Byte signed)
RdActRuntime	read actual runtime in ms Par.4 = value (3 Bytes)

RdError	<p>read error</p> <p>Par.4 = value (2 Bytes)</p> <p>Byte 0: Error code</p> <p>0: no error 1: undefined error 2: communication error 3: prohibited parameter used 4: input disabled but used for motion 5: no REF found between LS1 and LS2 6: LS or REF move need to be done first when valid area was left 7: Stop instruction proceeded 8: EMSTOP is/was active 9: proceeded no. of steps did not match previewed 10: two or more parameter for motion profile do not fit together</p> <p>Byte 1: Warning code</p> <p>1: jerk values set to zero 2: Vmax never reached 3: step added at the end of motion profile to reach correct no. of steps 4: steps removed at the end of motion profile to reach correct no. of steps 5: Vstop to big ($V_{stop} > 2 \cdot V_{min}$) 6: Vmin was bigger than Vmax -> Vmin reduced to Vmax 7: one or more parameter for motion profile out of range 8: no motion parameters loaded after module reset</p>
RdFwVersion	<p>read firmware version</p> <p>Par.4 = value (4 Byte/Character)</p> <p>format x.yy</p>

Affichage de l'état sur les adresses de module

Si le PCD ne communique pas avec le module (pas de FB H222 actif), certains bits d'état doivent être directement lus en tant qu'entrées sur les adresses de module. Le LSB correspondant à l'adresse de base du module.

Bits d'état : accès RO

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	FlagsValid	x	x	x	Error Y	Error X	OnDest Y	OnDest X

MSB

LSB

4

OnDest X et Y : ces bits sont à 0 à partir du moment où une instruction de course est émise et jusqu'à ce qu'elle soit exécutée. Si une instruction de course est émise et attend le signal de déclenchement, le bit « on destination » est également à 0.

Error X et Y : si une erreur est détectée pour un axe, le bit Error est positionné.

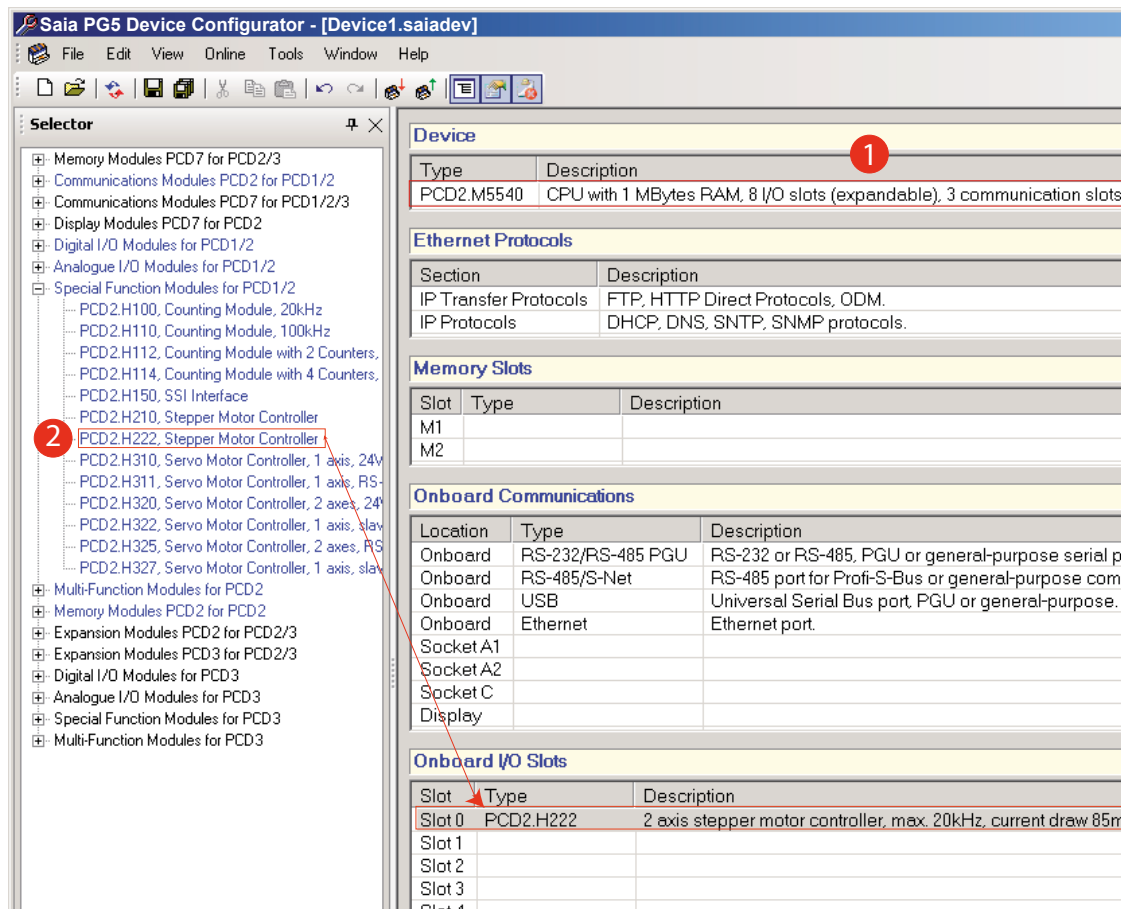
FlagsValid : ce bit signale si les indicateurs sont valides (1) ou pas (0) sur les adresses d'entrée 0 à 7.

4.2 Configuration du système avec le Configurateur d'appareil

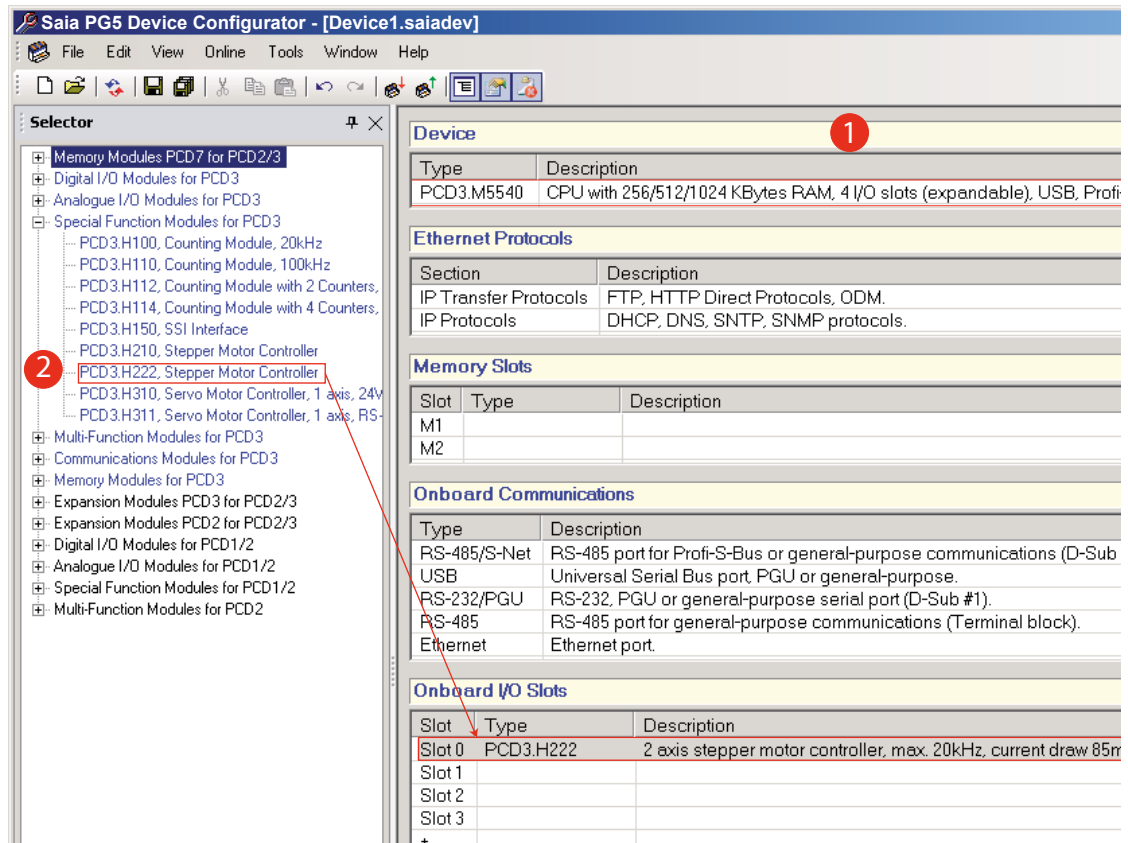
Pour commencer un nouveau projet, il faut d'abord sélectionner l'unité centrale (CPU) ❶ et le module concerné à l'aide du Configurateur d'appareil (device configurator). Le module se trouve sous l'onglet « Modules fonctionnels spéciaux pour PCD 1/2 » ou « Modules fonctionnels spéciaux pour PCD 3 » dans le volet de gauche ❷ et peut être amené sur le « slot » (emplacement) correspondant par « glisser-déposer ».

La fenêtre des propriétés qui permet de définir l'accès aux axes X et Y pour les entrées et les sorties se trouve sous l'onglet de droite. Une fonction de déclenchement peut être attribuée à une entrée ou à une sortie. Cette fonction de déclenchement spéciale permet de configurer le mouvement des axes.

4



Configurateur d'appareil : exemple pour le PCD2



4

Configurateur d'appareil : exemple pour le PCD3

4.3 Propriétés

La configuration programmée est affichée sous l'onglet de droite :

Cette configuration permet de définir la séquence des deux axes et la fonction de déclenchement. Ces propriétés permettent aussi d'activer un contacteur de fin de course ou un contacteur de référence, si nécessaire.

Properties

Slot 0 : PCD3.H222, Stepper Motor Controller

- General**
 - Base Address: 0
 - Connector Type: Type A, Spring Term
- Power Consumption**
 - Power Consumption 5V [mA]: 85
- Media Mapping Status**
 - Media Mapping For Status Enabled: No
 - Media Type For Status: Register
 - Number Of Media For Status: 4
 - Media Address For Status: 0
 - Base Symbol Definitions For Status: **S.IO.Slot0.Status**
 - Symbol Definitions For Status: **(Custom)**
- Media Mapping Position/Speed**
 - Media Type For Position/Speed: Register
 - Number Of Media For Position/Speed: 6
 - Media Address For Position/Speed: 0
 - Base Symbol Definitions For Position/Sp: **S.IO.Slot0.SpeedF**
 - Symbol Definitions For Position/Speed: **(Custom)**
- Media Mapping Outputs**
 - Media Mapping For Outputs Enabled: No
- X Axis**
 - Input LS1 Enabled For X Axis: **Yes**
 - Input LS1 Polarity For X Axis: Not Inverted
 - Input LS2 Enabled For X Axis: No
 - Input LS2 Polarity For X Axis: Not Inverted
 - Input REF Enabled For X Axis: No
 - Input REF Polarity For X Axis: Not Inverted
 - Input EMSTOP Enabled For X Axis: No
 - Input EMSTOP Polarity For X Axis: Not Inverted
 - Trigger IO For X Axis: Input
 - Trigger Enabled For X Axis: No
- Y Axis**
 - Input LS1 Enabled For Y Axis: No
 - Input LS1 Polarity For Y Axis: Not Inverted
 - LS2 Input Enabled For Y Axis: No
 - Input LS2 Polarity For Y Axis: Not Inverted
 - Input REF Enabled For Y Axis: No
 - Input REF Polarity For Y Axis: Not Inverted
 - Input EMSTOP Enabled For Y Axis: No
 - Input EMSTOP Polarity For Y Axis: Not Inverted
 - IO Trigger For Y Axis: Input

X Axis

4.4 Jeu d'instructions - Accès direct

4.4.1 Instructions d'écriture

Dimen- sions	Constante/variable :	Description :
O	IO.Slot0.IOAccess.X_AXIS_WR_START- MP	Initialiser et démarrer le mouvement : 0 : démarrer le profil de course actif cw 1 : démarrer le profil de course actif ccw
O	IO.Slot0.IOAccess.X_AXIS_WR_SING- LESTEP	Incrément unitaire avec Vmin : 0 : sens horaire 1 : sens antihoraire
O	IO.Slot0.IOAccess.X_AXIS_WR_MOVE- VCONST	Incréments à vitesse constante 0 : sens horaire 1 : sens antihoraire
O	IO.Slot0.IOAccess.X_AXIS_WR_LS- REFMOVE	Positionnement sur contacteur fin de course ou contacteur de référence : 1 : LS1 2 : LS2 3 : REF cw (recherche de LS1 ou REF dans le sens antihoraire) 4 : REF cw (recherche de LS2 ou REF dans le sens antihoraire) 5 : REF cw sur plage invalide sous LS1 (ccw d'abord) 6 : REF cw sur plage invalide sous LS2 (cw d'abord) 7 : REF cw (directement vers REF) 8 : REF ccw (directement vers REF)
O	IO.Slot0.IOAccess.X_AXIS_WR_STOP	Arrête le mouvement (cette instruction définit dans le registre des erreurs une erreur ARRET qui doit être acquit- tée par CONT avant la prochaine instruction de mouvement)
O	IO.Slot0.IOAccess.X_AXIS_WR_CONT	Acquitte dans le registre des erreurs l'erreur AR- RET qui a été provoquée par l'instruction d'arrêt.
M	IO.Slot0.IOAccess.X_AXIS_WR_VMIN	Charger la fréquence minimale en Hz : Plage : 10 à 10 000
M	IO.Slot0.IOAccess.X_AXIS_WR_VMAX	Charger la fréquence maximale en Hz : Plage : 20 à 20 000
M	IO.Slot0.IOAccess.X_AXIS_WR_ACC	Charger l'accélération en kHz/s : Plage : 1 à 1 000
M	IO.Slot0.IOAccess.X_AXIS_WR_DEC	Charger la décélération en kHz/s : Plage : 1 à 1 000
M	IO.Slot0.IOAccess.X_AXIS_WR_JER- KACC	Accélération - charger la valeur de jerk en pour- centage : Plage : 0 à 50
M	IO.Slot0.IOAccess.X_AXIS_WR_JERK- DEC	Décélération - charger la valeur de jerk en pour- centage : Plage : 0 à 50
O	IO.Slot0.IOAccess.X_AXIS_WR_SA- VEMP	Enregistrer le profil de course x (16 max.) : 0 : profil 0 1 : profil 1 ... 15 : profil 15
DM	IO.Slot0.IOAccess.X_AXIS_WR_DES- TREL	Charger la cible relative en incréments : Plage : 0 à 16 777 215 (3 octets) (mouvement continu si valeur 0xFFFFFFFF [16 777 215])

DM	IO.Slot0.IOAccess.X_AXIS_WR_DESTABS	Charger la cible en position absolue : Plage : -2 147 483 648 à +2 147 483 647 (4 octets) (la cible relative ne doit pas dépasser 16 777 215 incréments). WR_DESTABS doit être suivi de WR_MOVEV-CONST ou de WR_STARTMP. WR_STOP, WR_SINGLESTEP et WR_LSREFMOVE annulent cette instruction.
DM	IO.Slot0.IOAccess.X_AXIS_WR_ACTPOSABS	Définir la position absolue de la position d'axe instantanée : Plage : -2 147 483 648 à +2 147 483 647 (4 octets)
O	IO.Slot0.IOAccess.X_AXIS_WR_IOCONF	Charger la configuration des entrées : Bit 0 : M/A entrée LS1 (0 = Mar, 1 = Arr) Bit 1 : M/A entrée LS2 Bit 2 : M/A entrée REF Bit 3 : M/A entrée arrêt d'urgence Bit 4 : Entrée mode LS1 (0 = ouv. cont./1 = ferm. cont.) Bit 5 : Mode entrée LS2 Bit 6 : Mode entrée REF Bit 7 : Mode entrée arrêt d'urgence
O	IO.Slot0.IOAccess.X_AXIS_WR_TRIGCONF	Charger configuration E/S de déclenchement : Bit 0 : utiliser TRIG en entrée (0)/sortie (1) Bit 1 : utiliser TRIG (0 = TRIG inutilisé/1 = TRIG utilisé)
O	IO.Slot0.IOAccess.X_AXIS_WR_ACTMP	Charger le profil de course x (16 max.) : 0 : profil 0 1 : profil 1 ... 15 : profil 15

4.4.2 Instructions de lecture

Dimen- sions	Constante/variable :	Description :
O	IO.Slot0.IOAccess.X_AXIS_RD_IOSR	Bit 7 : réservé Bit 6 : état sortie MOTEN 0 : axe arrêté 1 : axe en mouvement Bit 5 : état sortie DIR 0 : l'axe tourne dans le sens horaire 1 : l'axe tourne dans le sens antihoraire Bit 4 : état de l'E/S TRIG Bit 3 : état de l'entrée EMSTOP Bit 2 : état de l'entrée REF Bit 1 : état de l'entrée LS2 Bit 0 : état de l'entrée LS1
O	IO.Slot0.IOAccess.X_AXIS_RD_SR	Bit 7 : 2 réservé Bit 1 : marquage cible atteinte 0 : hors position finale sur position finale Bit 0 : attente de TRIG 0 : positionnement activé, non démar- ré 1 : positionnement démarre si TRIG = 1
M	IO.Slot0.IOAccess.X_AXIS_RD_ERR_WARN	L'accès en lecture renvoie le dernier code d'erreur et d'avertissement depuis le module. Après lecture, le code est acquitté.
DM	IO.Slot0.IOAccess.X_AXIS_RD_SPEED	L'accès en lecture renvoie la vitesse instantanée (Hz).
DM	IO.Slot0.IOAccess.X_AXIS_RD_RUN-TIME	Renvoie le temps de positionnement actuel (depuis le début de l'ordre de course) en ms (3 octets)
DM	IO.Slot0.IOAccess.X_AXIS_RD_ACT-POSREL	L'accès en lecture renvoie le nombre d'incréments effectués depuis le début de l'ordre de course (3 octets).
M	IO.Slot0.IOAccess.X_AXIS_RD_ACT-POSABS	L'accès en lecture renvoie la position instantanée (4 octets).
M	IO.Slot0.IOAccess.X_AXIS_RD_VMIN	Renvoie la fréquence minimale en Hz.
M	IO.Slot0.IOAccess.X_AXIS_RD_VMAX	Renvoie la fréquence maximale en Hz.
M	IO.Slot0.IOAccess.X_AXIS_RD_ACC	Renvoie l'accélération en kHz/s.
M	IO.Slot0.IOAccess.X_AXIS_RD_DEC	Renvoie la décélération en kHz/s.
M	IO.Slot0.IOAccess.X_AXIS_RD_JER-KACC	Renvoie l'accélération en pourcentage.
M	IO.Slot0.IOAccess.X_AXIS_RD_JERK-DEC	Renvoie la décélération en pourcentage.
DM	IO.Slot0.IOAccess.X_AXIS_RD_DEST-REL	Renvoie le nombre d'incréments utilisés pour les instructions de course (3 octets).
DM	IO.Slot0.IOAccess.X_AXIS_RD_DE-STABS	Renvoie la position cible absolue utilisée pour la prochaine instruction de course (4 octets).
O	IO.Slot0.IOAccess.X_AXIS_RD_IOCONF	Configuration des entrées (description, voir WR_IOCONF)
O	IO.Slot0.IOAccess.X_AXIS_RD_TRIG-CONF	Configuration de la connexion TRIG (description, voir WR_TRIGCONFIG)
DM	IO.Slot0.IOAccess.RD_MODULEFW-VERSION	Renvoie la version du microprogramme au format x.yy ASCII.

De plus amples informations à ce sujet sont disponibles au chapitre 4.3.2. « Instructions de lecture ».



En fonction du câblage et de l'état de LS1 et LS2, les voyants 1 et 4 peuvent s'allumer en orange.

5 Adaptations du programme en cas de remplacement d'un module H210 par un module H222

Vous trouverez ci-dessous les différences les plus importantes au niveau du programme en cas de remplacement d'un module H210 par un module H222 :

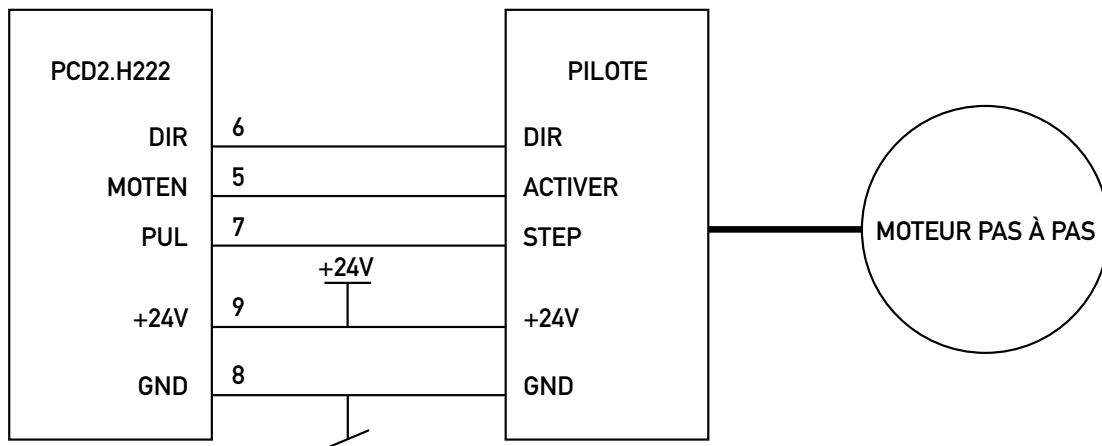
- H222.Init initialise non seulement le module, mais aussi le profil de course.
→ H222.Init et H222.InitMP sont nécessaires pour mettre l'axe en mouvement.
- Pas de sélection de la plage de fréquence → Simplification d'Init
- Vmin, Vmax et Acceleration ne dépendent plus de la plage de fréquences, mais doivent être directement interprétés comme valeur de fréquence.
→ Une conversion est impérative.
- Les contacteurs d'arrêt d'urgence, de fin de course et de référence doivent être activés à l'aide de nouvelles commandes.
- La course de référence n'est pas démarrée à l'aide d'H210.Home, mais d'H222.Exec et du paramètre de course de référence correspondant. Les commandes H222.Exec et le nombre de paramètres ne sont pas identiques avec les commandes correspondant à H210.Exec
→ toutes les commandes doivent être vérifiées.

6 Exemples

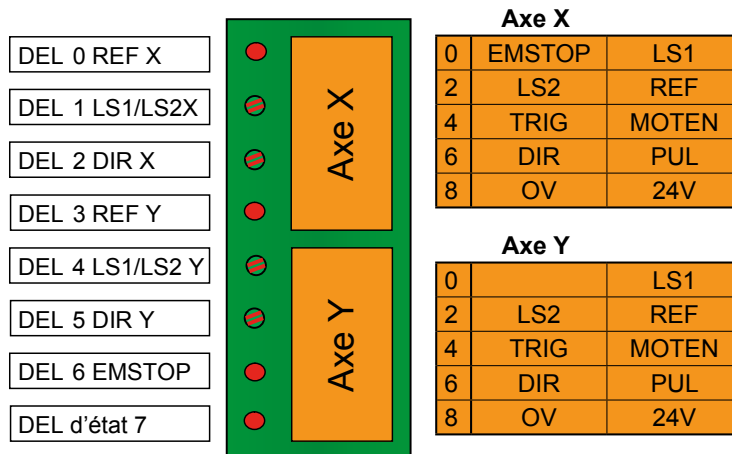
6.1 Matériel PCD2

- PCD2.H222
- PCD2.M5540
- Etage de sortie moteur pas à pas (driver)

Cet exemple montre une application simple du module PCD2.H222 :



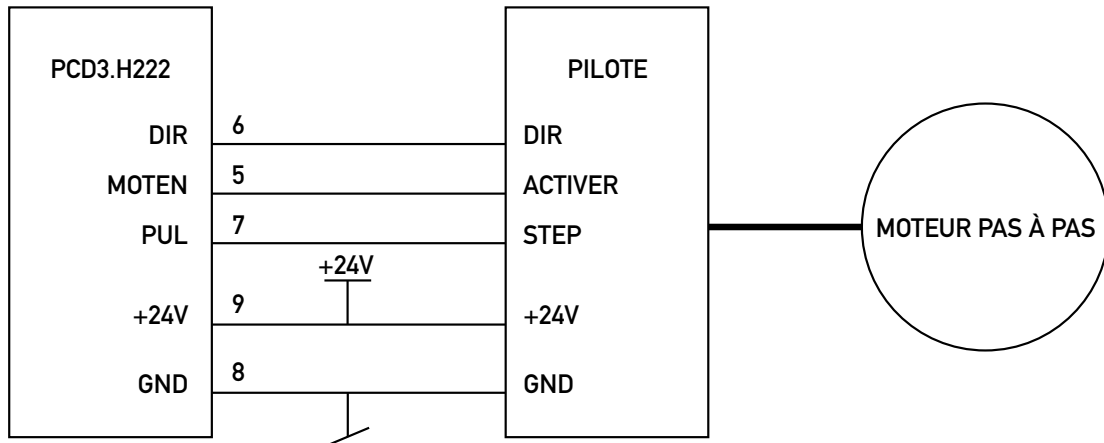
6



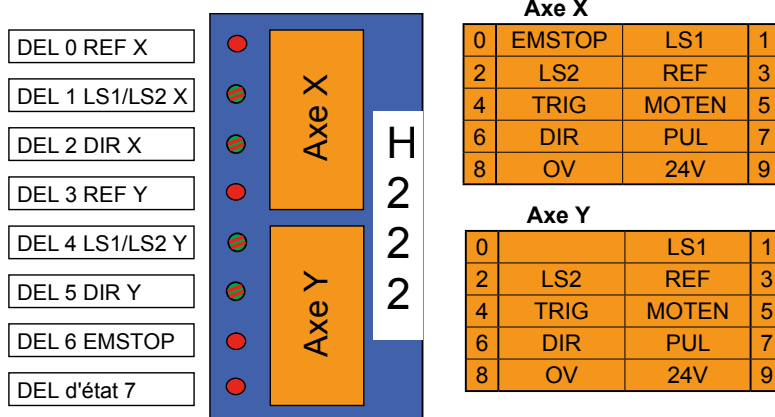
6.2 Matériel PCD3

- PCD3.H222
- PCD3.M5340
- Etage de sortie moteur pas à pas (driver)

Cet exemple montre une application simple du module PCD3.H222 :



6



6.3 Description du programme

Dans la première partie du programme IL sont définis divers registres qui seront utilisés lors de son exécution.

Le XOB 16 définit les valeurs d'un profil de course et les écrit dans le module H222 via l'emplacement d'E/S 0.

Dans le programme principal COB 0, la présence de l'axe dans sa position finale est vérifiée. Dès que cette condition est remplie, un nouvel ordre de mouvement est lancé selon le profil de course défini, le nombre d'incréments en « rPosition » et le sens de rotation en « rDir ».

6.4 Code du programme IL

6

Définitions

rUmin	EQU	R	; minimal speed
rUmax	EQU	R	; maximal speed
rAccel	EQU	R	; acceleration value
rDecel	EQU	R	; deceleration value
rJerkAcc	EQU	R	; jerk vaule for acceleration
rJerkDec	EQU	R	; jerk vaule for deceleration
rPosition	EQU	R	; number of steps to move
rDir	EQU	R	; 0=Vorwärts, 1=Rückwärts
rStatus	EQU	R	

Initialisation

```

;*****
;THIS IS A DEMONSTRATION PROGRAM FOR STEPPING MOTOR CONTROLLER
;
      XOB      16

; definiere Motionprofil-Werte X-Achse

      LD      rUmin      ; Umin: 100Hz
      100
      LD      rUmax      ; Umax: 1kHz
      1000
      LD      rAccel     ; acceleration: 100kHz/s
      100
      LD      rDecel     ; deceleration: 10kHz/s
      10
      LD      rJerkAcc   ; max. S-Shape in acceleration
      50
      LD      rJerkDec   ; no S-Shape in deceleration
      0
      LD      rPosition  ; run 5000 steps
      5000
      LD      rDir       ; direction: clockwise
      0

; configure module in Slot0, axis x
WRPW  IO.Slot0.IOAccess.X_AXIS_WR_UMIN
      rUmin
WRPW  IO.Slot0.IOAccess.X_AXIS_WR_UMAX
      rUmax
WRPW  IO.Slot0.IOAccess.X_AXIS_WR_ACC
      rAccel
WRPW  IO.Slot0.IOAccess.X_AXIS_WR_DEC
      rDecel
WRPW  IO.Slot0.IOAccess.X_AXIS_WR_JERKACC
      rJerkAcc
WRPW  IO.Slot0.IOAccess.X_AXIS_WR_JERKDEC
      rJerkDec

; configure module in Slot0, axis y
WRPW  IO.Slot0.IOAccess.Y_AXIS_WR_UMIN
      rUmin
WRPW  IO.Slot0.IOAccess.Y_AXIS_WR_UMAX
      rUmax
WRPW  IO.Slot0.IOAccess.Y_AXIS_WR_ACC
      rAccel
WRPW  IO.Slot0.IOAccess.Y_AXIS_WR_DEC
      rDecel
WRPW  IO.Slot0.IOAccess.Y_AXIS_WR_JERKACC
      rJerkAcc
WRPW  IO.Slot0.IOAccess.Y_AXIS_WR_JERKDEC
      rJerkDec

      EXOB

```

```

-----
;*****
      COB      0
              0

; check and start module in Slot0, axis x
RDPB  IO.Slot0.IOAccess.X_AXIS_RD_SR
      rStatus      ; read modul status register
SHIR  rStatus      ; OnDest flag in Accu
      2
JR    L Y_READY    ; if axis x still in move, jump to Y_READY

WRP   IO.Slot0.IOAccess.X_AXIS_WR_DESTREL
      rPosition    ; set number of steps to go
WRPB  IO.Slot0.IOAccess.X_AXIS_WR_STARTMP ; start motion profile
      rDir         ; set direction (0: clockwise, 1: conterclockwise

; check and start module in Slot0, axis y
Y_READY: RDPB  IO.Slot0.IOAccess.Y_AXIS_RD_SR
      rStatus      ; read modul status register
SHIR  rStatus      ; OnDest flag in Accu
      2
JR    L END_COB    ; if axis y still in move, jump to END_COB






WRP   IO.Slot0.IOAccess.Y_AXIS_WR_DESTREL
      rPosition    ; set number of steps to go
WRPB  IO.Slot0.IOAccess.Y_AXIS_WR_STARTMP ; start motion profile
      rDir         ; set direction (0: clockwise, 1: conterclockwise

END_COB:  ECOB

```

A Annexe

A.1 Icones

	Ce symbole informe le lecteur de la présence d'informations supplémentaires dans le présent manuel, dans un autre manuel dans une brochure technique. Il n'existe en général aucun lien direct vers ces documents.
	Ce symbole informe le lecteur d'un risque de décharge électrique en cas de contact. Recommandation : avant de toucher un composant électronique, vous devez au moins toucher préalablement la borne négative du système (boîtier de la douille PGU). Il est préférable d'être relié en permanence à la borne négative au moyen d'un bracelet anti-statique porté au poignet.
	Ce symbole signale des instructions qui doivent être strictement observées.
	Les explications accompagnant ce symbole s'appliquent uniquement à la série Classic de Saia-Burgess Controls AG.
	Les explications accompagnant ce symbole s'appliquent uniquement à la série PCD xx7 de Saia-Burgess Controls AG.

A.2 Adresse de la société Saia-Burgess Controls AG**Saia-Burgess Controls AG**

Bahnhofstrasse 18
3280 Murten / Suisse

Téléphone : +41 26 580 30 00

Télécopie : +41 26 580 34 99

E-mail : support@saia-pcd.com

Page d'accueil : www.saia-pcd.com

Assistance: www.sbc-support.ch

Entreprises de distribution international &

Représentants SBC : www.saia-pcd.com/contact

**Adresse postale pour les retours de produits
par les clients de "Vente Suisse" :****Saia-Burgess Controls AG**

Service Après-Vente
Rue de la Gare 18
CH-3280 Morat / Suisse