

Modicon M221 Logic Controller

Guide de la bibliothèque des fonctions avancées

Traduction de la notice originale

EIO0000003306.03

09/2025

Mentions légales

Les informations fournies dans ce document contiennent des descriptions générales, des caractéristiques techniques et/ou des recommandations concernant des produits/solutions.

Ce document n'est pas destiné à remplacer une étude détaillée ou un plan de développement ou de représentation opérationnel et propre au site. Il ne doit pas être utilisé pour déterminer l'adéquation ou la fiabilité des produits/solutions pour des applications utilisateur spécifiques. Il incombe à chaque utilisateur individuel d'effectuer, ou de faire effectuer par un professionnel de son choix (intégrateur, spécificateur ou équivalent), l'analyse de risques exhaustive appropriée ainsi que l'évaluation et les tests des produits/solutions par rapport à l'application ou l'utilisation particulière envisagée.

La marque Schneider Electric et toutes les marques de commerce de Schneider Electric SE et de ses filiales mentionnées dans ce document sont la propriété de Schneider Electric SE ou de ses filiales. Toutes les autres marques peuvent être des marques de commerce de leurs propriétaires respectifs.

Ce document et son contenu sont protégés par les lois sur la propriété intellectuelle applicables et sont fournis à titre d'information uniquement. Aucune partie de ce document ne peut être reproduite ou transmise sous quelque forme ou par quelque moyen que ce soit (électronique, mécanique, photocopie, enregistrement ou autre), à quelque fin que ce soit, sans l'autorisation écrite préalable de Schneider Electric.

Schneider Electric n'accorde aucun droit ni aucune licence d'utilisation commerciale de ce document ou de son contenu, sauf dans le cadre d'une licence non exclusive et personnelle, pour le consulter tel quel.

Schneider Electric se réserve le droit d'apporter à tout moment des modifications ou des mises à jour relatives au contenu de ce document ou à son format, sans préavis.

Dans la mesure permise par la loi applicable, Schneider Electric et ses filiales déclinent toute responsabilité en cas d'erreurs ou d'omissions dans le contenu informatif du présent document ou pour toute conséquence résultant de l'utilisation des informations qu'il contient.

Table des matières

Consignes de sécurité	7
Avant de commencer	7
Démarrage et test	8
Fonctionnement et réglages	9
A propos de ce document.....	10
Introduction aux fonctions avancées	15
Introduction.....	16
E/S expertes.....	17
Mappage d'E/S expertes intégrées.....	19
Informations générales sur la gestion des blocs fonction	21
Fonctions avancées des entrées expertes	22
Fast Counter (%FC).....	23
Description	23
Configuration.....	24
Exemple de programmation.....	26
High Speed Counter (%HSC)	28
Description	28
Compteur HSC et modes de comptage.....	32
Compteur HSC en mode fréquencemètre	38
Fonctions avancées des sorties expertes.....	40
Bloc fonction Pulse (%PLS).....	41
Description	41
Configurer des blocs de fonctions	42
Exemple de programmation.....	47
Bloc fonction Pulse Width Modulation (%PWM)	48
Description	48
Configuration de bloc fonction.....	49
Exemple de programmation.....	52
Drive (%DRV)	53
Description	53
Etats du variateur et du contrôleur logique	55
Ajout d'un bloc fonction Drive	57
Configurer des blocs de fonctions	58
MC_Power_ATV : Activer/désactiver l'étage de puissance	58
MC_Jog_ATV : Démarrer le mode Jog.....	60
MC_MoveVel_ATV : Déplacement à une vitesse spécifiée	63
MC_Stop_ATV : Arrêt de mouvement	66
MC_ReadStatus_ATV : Lire l'état de l'équipement	68
MC_ReadMotionState_ATV : Lire l'état de mouvement	70
MC_Reset_ATV : Acquiescement et réinitialisation d'erreur	72
Codes d'erreur.....	75
Bloc fonction Pulse Train Output (%PTO)	78
Description	78
PTO (Pulse Train Output, sortie à train d'impulsions).....	78
Modes de sortie d'impulsion	80
Rampes d'accélération et de décélération	81
Événement de capteur.....	83
Compensation de jeu.....	85

Limites de positionnement	86
Configuration.....	88
Configuration de la fonction PTO	88
Motion Task Table	88
Programmation.....	95
Ajout/suppression d'un bloc fonction	95
Blocs fonction PTO.....	96
Modes de référencement.....	97
Modes de référencement	97
Définition de position	99
Référence longue.....	99
Référence courte sans inversion	100
Référence courte avec inversion	101
Décalage d'origine	103
Paramètres des données.....	103
Codes d'objet de bloc fonction.....	103
Modes de fonctionnement	108
Diagramme d'état de mouvement.....	108
Mode de tampon	109
Blocs fonction de mouvement	111
Bloc fonction <i>MC_MotionTask_PTO</i>	111
Bloc fonction <i>MC_Power_PTO</i>	114
Bloc fonction <i>MC_MoveVel_PTO</i>	117
Bloc fonction <i>MC_MoveRel_PTO</i>	120
Bloc fonction <i>MC_MoveAbs_PTO</i>	123
Bloc fonction <i>MC_Home_PTO</i>	126
Bloc fonction <i>MC_SetPos_PTO</i>	128
Bloc fonction <i>MC_Stop_PTO</i>	130
Bloc fonction <i>MC_Halt_PTO</i>	132
Blocs fonction d' administration	134
Bloc fonction <i>MC_ReadActVel_PTO</i>	134
Bloc fonction <i>MC_ReadActPos_PTO</i>	136
Bloc fonction <i>MC_ReadSts_PTO</i>	137
Bloc fonction <i>MC_ReadMotionState_PTO</i>	139
Bloc fonction <i>MC_ReadAxisError_PTO</i>	140
Bloc fonction <i>MC_Reset_PTO</i>	142
Bloc fonction <i>MC_TouchProbe_PTO</i>	143
Bloc fonction <i>MC_AbortTrigger_PTO</i>	145
Bloc fonction <i>MC_ReadPar_PTO</i>	146
Bloc fonction <i>MC_WritePar_PTO</i>	147
Générateur de fréquence (%FREQGEN).....	149
Description	149
Configuration.....	151
Fonctions logicielles avancées	153
Fonction PID	154
Modes de marche des PID	154
Modes de fonctionnement des PID	154
Configuration Auto-Tuning du PID	155
Configuration d'Auto-Tuning du PID.....	155
Configuration standard du PID	158
Configuration de l'adresse de mot du PID.....	158

Réglage du PID avec la fonction Auto-Tuning (AT).....	161
Mode Manuel.....	165
Détermination de la période d'échantillonnage (Ts)	166
Assistant PID.....	168
Accès à l'Assistant PID	168
Onglet Général	169
Onglet Entrée	171
Onglet PID.....	172
Onglet AT	174
Onglet Sortie.....	175
Programmation du PID.....	177
Description	177
Programmation et configuration.....	178
Etats de PID et codes d'erreur détectée	179
Annexes	182
Paramètres du PID	183
Rôle et influence des paramètres PID.....	183
Méthode de réglage des paramètres PID	184
Glossaire	187
Index	189

Consignes de sécurité

Informations importantes

Lisez attentivement ces instructions et examinez le matériel pour vous familiariser avec l'appareil avant de tenter de l'installer, de le faire fonctionner, de le réparer ou d'assurer sa maintenance. Les messages spéciaux suivants que vous trouverez dans cette documentation ou sur l'appareil ont pour but de vous mettre en garde contre des risques potentiels ou d'attirer votre attention sur des informations qui clarifient ou simplifient une procédure.



La présence de ce symbole sur une étiquette "Danger" ou "Avertissement" signale un risque d'électrocution qui provoquera des blessures physiques en cas de non-respect des consignes de sécurité.



Ce symbole est le symbole d'alerte de sécurité. Il vous avertit d'un risque de blessures corporelles. Respectez scrupuleusement les consignes de sécurité associées à ce symbole pour éviter de vous blesser ou de mettre votre vie en danger.

⚠ DANGER
DANGER signale un risque qui, en cas de non-respect des consignes de sécurité, provoque la mort ou des blessures graves.
⚠ AVERTISSEMENT
AVERTISSEMENT signale un risque qui, en cas de non-respect des consignes de sécurité, peut provoquer la mort ou des blessures graves.
⚠ ATTENTION
ATTENTION signale un risque qui, en cas de non-respect des consignes de sécurité, peut provoquer des blessures légères ou moyennement graves.
AVIS
AVIS indique des pratiques n'entraînant pas de risques corporels.

Remarque Importante

L'installation, l'utilisation, la réparation et la maintenance des équipements électriques doivent être assurées par du personnel qualifié uniquement. Schneider Electric décline toute responsabilité quant aux conséquences de l'utilisation de ce matériel.

Une personne qualifiée est une personne disposant de compétences et de connaissances dans le domaine de la construction, du fonctionnement et de l'installation des équipements électriques, et ayant suivi une formation en sécurité leur permettant d'identifier et d'éviter les risques encourus.

Avant de commencer

N'utilisez pas ce produit sur les machines non pourvues de protection efficace du point de fonctionnement. L'absence de ce type de protection sur une machine présente un risque de blessures graves pour l'opérateur.

▲ AVERTISSEMENT

EQUIPEMENT NON PROTEGE

- N'utilisez pas ce logiciel ni les automatismes associés sur des appareils non équipés de protection du point de fonctionnement.
- N'accédez pas aux machines pendant leur fonctionnement.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Cet automatisme et le logiciel associé permettent de commander des processus industriels divers. Le type ou le modèle d'automatisme approprié pour chaque application dépendra de facteurs tels que la fonction de commande requise, le degré de protection exigé, les méthodes de production, des conditions inhabituelles, la législation, etc. Dans certaines applications, plusieurs processeurs seront nécessaires, notamment lorsque la redondance de sauvegarde est requise.

Vous seul, en tant que constructeur de machine ou intégrateur de système, pouvez connaître toutes les conditions et facteurs présents lors de la configuration, de l'exploitation et de la maintenance de la machine, et êtes donc en mesure de déterminer les équipements automatisés, ainsi que les sécurités et verrouillages associés qui peuvent être utilisés correctement. Lors du choix de l'automatisme et du système de commande, ainsi que du logiciel associé pour une application particulière, vous devez respecter les normes et réglementations locales et nationales en vigueur. Le document National Safety Council's Accident Prevention Manual (reconnu aux Etats-Unis) fournit également de nombreuses informations utiles.

Dans certaines applications, telles que les machines d'emballage, une protection supplémentaire, comme celle du point de fonctionnement, doit être fournie pour l'opérateur. Elle est nécessaire si les mains ou d'autres parties du corps de l'opérateur peuvent entrer dans la zone de point de pincement ou d'autres zones dangereuses, risquant ainsi de provoquer des blessures graves. Les produits logiciels seuls, ne peuvent en aucun cas protéger les opérateurs contre d'éventuelles blessures. C'est pourquoi le logiciel ne doit pas remplacer la protection de point de fonctionnement ou s'y substituer.

Avant de mettre l'équipement en service, assurez-vous que les dispositifs de sécurité et de verrouillage mécaniques et/ou électriques appropriés liés à la protection du point de fonctionnement ont été installés et sont opérationnels. Tous les dispositifs de sécurité et de verrouillage liés à la protection du point de fonctionnement doivent être coordonnés avec la programmation des équipements et logiciels d'automatisation associés.

NOTE: La coordination des dispositifs de sécurité et de verrouillage mécaniques/électriques du point de fonctionnement n'entre pas dans le cadre de cette bibliothèque de blocs fonction, du Guide utilisateur système ou de toute autre mise en œuvre référencée dans la documentation.

Démarrage et test

Avant toute utilisation de l'équipement de commande électrique et des automatismes en vue d'un fonctionnement normal après installation, un technicien qualifié doit procéder à un test de démarrage afin de vérifier que l'équipement fonctionne correctement. Il est essentiel de planifier une telle vérification et d'accorder suffisamment de temps pour la réalisation de ce test dans sa totalité.

⚠ AVERTISSEMENT

RISQUES INHERENTS AU FONCTIONNEMENT DE L'EQUIPEMENT

- Assurez-vous que toutes les procédures d'installation et de configuration ont été respectées.
- Avant de réaliser les tests de fonctionnement, retirez tous les blocs ou autres cales temporaires utilisés pour le transport de tous les dispositifs composant le système.
- Enlevez les outils, les instruments de mesure et les débris éventuels présents sur l'équipement.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Effectuez tous les tests de démarrage recommandés dans la documentation de l'équipement. Conservez toute la documentation de l'équipement pour référence ultérieure.

Les tests logiciels doivent être réalisés à la fois en environnement simulé et réel

Vérifiez que le système entier est exempt de tout court-circuit et mise à la terre temporaire non installée conformément aux réglementations locales (conformément au National Electrical Code des Etats-Unis, par exemple). Si des tests diélectriques sont nécessaires, suivez les recommandations figurant dans la documentation de l'équipement afin d'éviter de l'endommager accidentellement.

Avant de mettre l'équipement sous tension :

- Enlevez les outils, les instruments de mesure et les débris éventuels présents sur l'équipement.
- Fermez le capot du boîtier de l'équipement.
- Retirez toutes les mises à la terre temporaires des câbles d'alimentation entrants.
- Effectuez tous les tests de démarrage recommandés par le fabricant.

Fonctionnement et réglages

Les précautions suivantes sont extraites du document NEMA Standards Publication ICS 7.1-1995 :

(En cas de divergence ou de contradiction entre une traduction et l'original anglais, le texte original en anglais prévaudra.)

- Malgré le soin apporté à la conception et à la fabrication de l'équipement ou au choix et à l'évaluation des composants, des risques subsistent en cas d'utilisation inappropriée de l'équipement.
- Il arrive parfois que l'équipement soit dérégulé accidentellement, entraînant ainsi un fonctionnement non satisfaisant ou non sécurisé. Respectez toujours les instructions du fabricant pour effectuer les réglages fonctionnels. Les personnes ayant accès à ces réglages doivent connaître les instructions du fabricant de l'équipement et les machines utilisées avec l'équipement électrique.
- L'opérateur ne doit avoir accès qu'aux réglages fonctionnels dont il a besoin. L'accès aux autres commandes doit être limité afin d'empêcher les changements non autorisés des caractéristiques de fonctionnement.

A propos de ce document

Domaine de validité du document

Ce document décrit les fonctions avancées EcoStruxure Machine Expert - Basic et leur rapport à la prise en charge de la fonction PID et des E/S expertes du M221 Logic Controller. Il détaille les fonctionnalités, caractéristiques et performances des fonctions avancées du M221 Logic Controller.

Note de validité

Ce document a été mis à jour pour la version EcoStruxure™ Machine Expert – Basic V1.4.

Informations spécifiques au produit

▲ AVERTISSEMENT

PERTE DE CONTROLE

- Réalisez une analyse des modes de défaillance et de leurs effets (FMEA) ou une analyse de risques équivalente sur l'application et appliquez les contrôles de prévention et de détection appropriés avant la mise en œuvre.
- Prévoyez un état de repli pour les événements ou séquences de commande indésirables.
- Le cas échéant, prévoyez des chemins de commande séparés et redondants.
- Définissez les paramètres appropriés, notamment pour les limites.
- Examinez les conséquences des retards de transmission et prenez les mesures correctives nécessaires.
- Examinez les conséquences des interruptions de la liaison de communication et prenez des mesures correctives nécessaires.
- Prévoyez des chemins indépendants pour les fonctions de commande critiques (arrêt d'urgence, dépassement de limites, conditions d'erreur, etc.) en fonction de votre évaluation des risques ainsi que des réglementations et consignes applicables.
- Appliquez les réglementations et les consignes locales de sécurité et de prévention des accidents.¹
- Testez chaque mise en œuvre d'un système pour vérifier son bon fonctionnement avant de le mettre en service.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

¹ Pour plus d'informations, consultez le document NEMA ICS 1.1 (dernière édition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* (Directives de sécurité pour l'application, l'installation et la maintenance de commande statique) et le document NEMA ICS 7.1 (dernière édition), *Safety Standards for Construction and Guide for Selection, Installation, and Operation of Adjustable-Speed Drive Systems* (Normes de sécurité relatives à la construction et manuel de sélection, d'installation et d'exploitation de variateurs de vitesse) ou leur équivalent en vigueur dans votre pays.

⚠ AVERTISSEMENT

FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

- N'utilisez que le logiciel approuvé par Schneider Electric pour faire fonctionner cet équipement.
- Mettez à jour votre programme d'application chaque fois que vous modifiez la configuration matérielle physique.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Informations relatives à la cybersécurité générale

Ces dernières années, le nombre croissant de machines en réseau et d'usines de production a entraîné une augmentation correspondante du potentiel de cybermenaces, telles que les accès non autorisés, les violations de données et les perturbations opérationnelles. Vous devez donc envisager toutes les mesures de cybersécurité possibles pour protéger les ressources et les systèmes contre de telles menaces.

Pour garantir la sécurité et la protection de vos produits Schneider Electric, il est dans votre intérêt d'appliquer les meilleures pratiques relatives à la cybersécurité telles que décrites dans le document *Cybersecurity Best Practices*.

Schneider Electric fournit des informations supplémentaires et une assistance :

- Abonnez-vous à la newsletter sur la sécurité de Schneider Electric.
- Consultez la page Web *Cybersecurity Support Portal* pour :
 - obtenir des notifications de sécurité.
 - signaler les vulnérabilités et incidents.
- Consultez la page Web *Schneider Electric Cybersecurity and Data Protection Posture* pour :
 - accéder à la position sur la cybersécurité.
 - en savoir plus sur la cybersécurité dans l'académie de cybersécurité.
 - découvrir les services de cybersécurité de Schneider Electric.

Langues disponibles du document

Ce document est disponible dans les langues suivantes :

- Anglais (EIO0000003305)
- Français (EIO0000003306)
- Allemand (EIO0000003307)
- Espagnol (EIO0000003308)
- Italien (EIO0000003309)
- Chinois (EIO0000003310)
- Portugais (EIO0000003311)
- Turc (EIO0000003312)

Document(s) à consulter

Titre du document	Numéro de référence
EcoStruxure Machine Expert - Basic - Guide d'utilisation	EIO0000003281 (ENG)
	EIO0000003282 (FRA)
	EIO0000003283 (GER)
	EIO0000003284 (SPA)
	EIO0000003285 (ITA)
	EIO0000003286 (CHS)
	EIO0000003287 (POR)
EIO0000003288 (TUR)	
EcoStruxure Machine Expert - Basic - Guide de la bibliothèque des fonctions génériques	EIO0000003289 (ENG)
	EIO0000003290 (FRA)
	EIO0000003291 (GER)
	EIO0000003292 (SPA)
	EIO0000003293 (ITA)
	EIO0000003294 (CHS)
	EIO0000003295 (POR)
EIO0000003296 (TUR)	
Modicon M221 Logic Controller - Guide de programmation	EIO0000003297 (ENG)
	EIO0000003298 (FRE)
	EIO0000003299 (GER)
	EIO0000003300 (SPA)
	EIO0000003301 (ITA)
	EIO0000003302 (CHS)
	EIO0000003304 (TUR)
EIO0000003303 (POR)	
Modicon M221 Logic Controller - Guide de référence du matériel	EIO0000003313 (ENG)
	EIO0000003314 (FRA)
	EIO0000003315 (GER)
	EIO0000003316 (SPA)
	EIO0000003317 (ITA)
	EIO0000003318 (CHS)
	EIO0000003319 (POR)
EIO0000003320 (TUR)	

Pour rechercher des documents en ligne, visitez le centre de téléchargement Schneider Electric (www.se.com/ww/en/download/).

Informations concernant la terminologie inclusive/sensible

Schneider Electric s'efforce de mettre constamment à jour ses communications et ses produits pour respecter ses engagements en matière de terminologie inclusive/sensible. Il se peut malgré tout que nos contenus présentent encore des termes jugés inappropriés par certains clients.

Terminologie utilisée dans les normes

Les termes techniques, la terminologie, les symboles et les descriptions correspondantes employés dans ce manuel ou figurant sur les produits eux-mêmes proviennent généralement des normes internationales.

Dans le domaine des systèmes de sécurité fonctionnelle, des variateurs et de l'automatisme en général, il s'agit par exemple de termes tels que *sécurité*, *fonction de sécurité*, *état sécurisé*, *défaut*, *réinitialisation de défaut*, *dysfonctionnement*, *panne*, *erreur*, *message d'erreur*, *dangereux*, etc.

Ces normes incluent entre autres les éléments suivants :

Norme	Description
IEC 61131-2:2007	Automates programmables, partie 2 : Spécifications et essais des équipements.
ISO 13849-1:2023	Sécurité des machines : Composants liés à la sécurité dans les systèmes de commande. Principes généraux de conception
EN 61496-1:2020	Sécurité des machines : Equipement de protection électrosensible. Partie 1 : Exigences générales et tests.
ISO 12100:2010	Sécurité des machines - Principes généraux de conception - Appréciation du risque et réduction du risque
EN 60204-1:2006	Sécurité des machines - Equipement électrique des machines - Partie 1 : exigences générales
ISO 14119:2013	Sécurité des machines - Dispositifs de verrouillage associés à des protecteurs - Principes de conception et de choix
ISO 13850:2015	Sécurité des machines - Fonction d'arrêt d'urgence - Principes de conception
IEC 62061:2021	Sécurité des machines - Sécurité fonctionnelle des systèmes de commande électrique, électronique et électronique programmables relatifs à la sécurité
IEC 61508-1:2010	Sécurité fonctionnelle des systèmes électriques, électroniques et électroniques programmables liés à la sécurité : Exigences générales.
IEC 61508-2:2010	Sécurité fonctionnelle des systèmes électriques, électroniques et électroniques programmables liés à la sécurité : Exigences concernant la sécurité fonctionnelle des systèmes électriques, électroniques et électroniques programmables liés à la sécurité.
IEC 61508-3:2010	Sécurité fonctionnelle des systèmes électriques, électroniques et électroniques programmables liés à la sécurité : Configuration logicielle requise.
IEC 61784-3:2021	Réseaux de communication industriels - Profils - Partie 3 : Bus de terrain liés à la sécurité fonctionnelle - Règles générales et définitions de profil.
2006/42/EC	Directive Machines
2014/30/EU	Directive sur la compatibilité électromagnétique
2014/35/EU	Directive sur les basses tensions

De plus, des termes utilisés dans le présent document peuvent provenir d'autres normes telles que :

Norme	Description
Série IEC 60034	Machines électriques rotatives
Série IEC 61800	Entraînements électriques de puissance à vitesse variable
Série IEC 61158	Communications numériques pour les systèmes de mesure et de commande – Bus de terrain utilisés dans les systèmes de commande industriels

Enfin, le terme *zone de fonctionnement* peut être utilisé dans le contexte de la description de dangers spécifiques et a la même signification que *zone à risque* ou *zone dangereuse* dans la directive *Machines (2006/42/EC)* et *ISO 12100:2010*.

NOTE: Les normes susmentionnées peuvent s'appliquer ou pas aux produits cités dans la présente documentation. Pour plus d'informations sur chacune des normes applicables aux produits décrits dans le présent document, consultez les tableaux de caractéristiques de ces références de produit.

Introduction aux fonctions avancées

Contenu de cette partie

Introduction.....	16
-------------------	----

Présentation

Cette section présente les différentes fonctions avancées, leurs modes disponibles, leurs fonctionnalités et leurs performances.

Introduction

Contenu de ce chapitre

E/S expertes	17
Mappage d'E/S expertes intégrées	19
Informations générales sur la gestion des blocs fonction	21

Présentation

Ce document décrit les fonctions avancées EcoStruxure Machine Expert - Basic et leur rapport à la prise en charge de la fonction PID et des E/S expertes du M221. Il détaille les fonctionnalités, caractéristiques et performances des E/S des fonctions Fast Counter (%FC), High Speed Counter (%HSC), Pulse (%PLS), Pulse Width Modulation (%PWM) et Pulse Train Output (%PTO). De plus, vous y trouverez une description complète de la fonctionnalité logicielle avancée PID. Pour plus d'informations sur les fonctions définies par l'utilisateur et les blocs fonction définis par l'utilisateur, consultez les sections Fonctions définies par l'utilisateur (voir EcoStruxure Machine Expert - Basic, Guide d'utilisation) et Blocs fonction définis par l'utilisateur (voir EcoStruxure Machine Expert - Basic, Guide d'utilisation).

Ces fonctions fournissent des solutions simples et puissantes à votre application. Toutefois, l'utilisation et l'application des informations fournies dans le présent document exigent des compétences en conception et en programmation des systèmes de commande automatisés.

Vous seul, en tant que constructeur ou intégrateur de machine, pouvez connaître toutes les conditions et facteurs présents lors de l'installation, de la configuration, de l'exploitation et de la maintenance de la machine ou des processus liés, et êtes donc en mesure de déterminer les équipements et systèmes d'automatisme, ainsi que les sécurités et verrouillages associés qui peuvent être utilisés correctement et efficacement. Pour choisir des équipements d'automatisation et de contrôle, ainsi que les autres équipements ou logiciels associés, pour une application spécifique, vous devez aussi prendre en compte les normes et réglementations locales, régionales ou nationales applicables.

▲ AVERTISSEMENT

INCOMPATIBILITÉ RÉGLEMENTAIRE

Assurez-vous que tous les équipements concernés et les systèmes conçus sont conformes à toutes les normes et réglementations locales, régionales et nationales applicables.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Les fonctionnalités offertes par les fonctions avancées EcoStruxure Machine Expert - Basic des contrôleurs M221 supposent que vous intégrez le matériel de sécurité nécessaire dans l'architecture de votre application, notamment des détecteurs de limites appropriés, des dispositifs d'arrêt d'urgence et des circuits de contrôle. Vous êtes implicitement tenu d'implémenter des mesures de sécurité fonctionnelle dans la conception de votre machine en vue d'éviter des comportements indésirables tels que les dépassements de fin de course ou toute autre forme de mouvement incontrôlé. Vous êtes également censé avoir effectué une analyse de sécurité fonctionnelle et une évaluation des risques convenables pour votre machine ou processus.

▲ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

Assurez-vous qu'une évaluation des risques est effectuée et respectée conformément à la norme EN/ISO 12100 pendant la conception de votre machine.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

E/S expertes

Introduction

Le M221 Logic Controller fournit :

- Quatre entrées rapides (%I0.0, %I0.1, %I0.6 et %I0.7)
- Deux sorties rapides sur les contrôleurs équipés de sorties transistor (%Q0.0 et %Q0.1)
- Quatre sorties rapides sur les contrôleurs de référence TM221C40U et TM221CE40U (%Q0.0, %Q0.1, %Q0.2 et %Q0.3)

NOTE: Les fonctions de sortie rapide ne sont pas prises en charge par les contrôleurs munis de sorties relais.

Le M221 Logic Controller prend en charge les fonctions d'E/S expertes (selon la référence du modèle) :

Fonctions		Description
Compteurs	Compteur rapide, page 23	La fonction FC peut effectuer des comptages rapides d'impulsions émises par des capteurs, des commutateurs, etc.
	Compteur HSC, page 28	La fonction HSC peut effectuer des comptages rapides d'impulsions émises par des capteurs, des commutateurs, etc., qui sont connectés aux entrées rapides.
Générateurs d'impulsions	Impulsion, page 41	La fonction PLS génère un signal d'impulsion à ondes carrées sur les voies de sortie dédiées.
	Modulation de largeur d'impulsion, page 48	La fonction PWM génère un signal d'onde modulée sur les voies de sortie dédiées avec un cycle de service variable.
	Sortie à train d'impulsions, page 78	La fonction PTO génère une sortie à train d'impulsions pour contrôler un moteur pas à pas ou un variateur monoaxe linéaire en mode Boucle ouverte.
	Générateur de fréquence, page 149	La fonction FREQGEN génère un signal carré sur une voie de sortie dédiée avec fréquence programmable et cycle de service de 50 %.

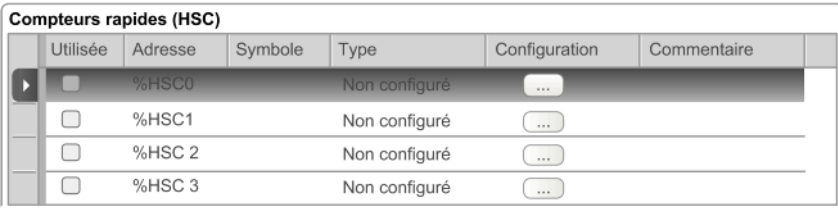
NOTE:

- Lorsqu'une entrée est utilisée comme entrée marche/arrêt (Run/Stop), elle ne peut pas être utilisée par une fonction experte.
- Lorsqu'une sortie est utilisée comme sortie d'alarme (Alarm), elle ne peut pas être utilisée par une fonction experte.

Pour plus d'informations, reportez-vous à la section Configuration des entrées/sorties intégrées (voir Modicon M221 Logic Controller - Guide de programmation).

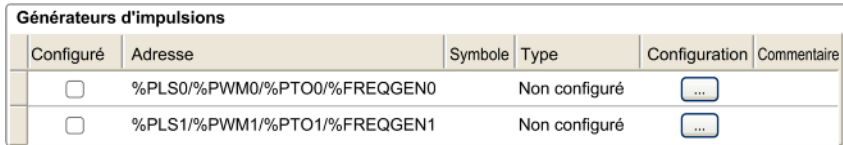
Configuration d'une fonction d'entrée experte

Pour configurer une fonction d'entrée experte, procédez comme suit :

Étape	Description
1	<p>Cliquez sur le nœud Compteurs rapides (HSC) dans l'arborescence du matériel. Résultat : La liste des compteurs HSC s'affiche :</p> 
2	<p>Cliquez sur ... dans la colonne Configuration pour sélectionner le type de compteur HSC et afficher la fenêtre Assistant HSC.</p>

Configuration d'une fonction de sortie experte

Pour configurer une fonction de sortie experte, procédez comme suit :

Étape	Description
1	<p>Cliquez sur le nœud Générateurs d'impulsions dans l'arborescence du matériel. Résultat : La liste Générateurs d'impulsions s'affiche :</p> 
2	<p>Cliquez sur [...] dans la colonne Configuration pour sélectionner le type de générateur d'impulsions et afficher la fenêtre Assistant pour les sorties à train d'impulsions.</p>

Caractéristiques de la configuration des fonctions d'E/S expertes

- Les entrées peuvent être lues dans les variables mémoire standard, même si elles sont configurées en association avec des fonctions d'E/S expertes.
- La gestion des courts-circuits continue à s'appliquer à toutes les sorties expertes.
- Toutes les E/S non utilisées par des fonctions d'E/S expertes sont utilisables comme des E/S normales.
- Les sorties utilisées par les fonctions *Pulse*, *Pulse Train Output*, *Pulse Width Modulation* et *High Speed Counters* ne sont accessibles qu'avec le bloc fonction d'E/S expertes. Elles ne peuvent pas être lues ni écrites directement dans l'application.

Mappage d'E/S expertes intégrées

Mappage d'entrées pour les fonctions expertes sur M221 Logic Controller

Les entrées numériques intégrées peuvent être affectées à des fonctions (Run/ Stop, Mémorisation, Événement, FC, HSC, PTO). Les entrées non affectées à des fonctions sont considérées comme normales. Le tableau suivant indique les affectations possibles des entrées numériques intégrées du M221 Logic Controller :

Fonction		Fonction d'entrée simple			Fonction d'entrée avancée		
		Run/Stop	Mémo- risa- tion	Evéne- ment	FC (Fast Counter)	HSC	PTO ⁽³⁾
Entrée rapide	%I0.0	X	-	-	-	%HSC0	-
	%I0.1	X	-	-	-	%HSC0 ou %HSC2 ⁽¹⁾	-
Entrée normale	%I0.2	X	X	X	%FC0	Présélection pour %HSC0	Ref or probe for %PTO0 to %PTO3
	%I0.3	X	X	X	%FC1	Capture pour %HSC0	
	%I0.4	X	X	X	%FC2	Capture pour %HSC1	
	%I0.5	X	X	X	%FC3	Présélection pour %HSC1	
Entrée rapide	%I0.6	X	-	-	-	%HSC1	-
	%I0.7	X	-	-	-	%HSC1 or %HSC3 ⁽²⁾	-
Entrée normale (selon la référence du contrôleur)	%I0.8	X	-	-	-	-	Référence ou sonde pour %PTO0 à %PTO3 sur les contrôleurs TM221C40U et TM221CE40U
	%I0.9	X	-	-	-	-	
	%I0.10	X	-	-	-	-	
	%I0.11	X	-	-	-	-	
	%I0.12	X	-	-	-	-	
	%I0.13	X	-	-	-	-	
	%I0.14	X	-	-	-	-	
	%I0.15	X	-	-	-	-	
	%I0.16	X	-	-	-	-	
	%I0.17	X	-	-	-	-	
	%I0.18	X	-	-	-	-	
	%I0.19	X	-	-	-	-	
	%I0.20	X	-	-	-	-	
%I0.21	X	-	-	-	-		
%I0.22	X	-	-	-	-		
%I0.23	X	-	-	-	-		

X Oui

- Non

⁽¹⁾ %HSC2 est disponible lorsque %HSC0 est configuré en tant que *Monophasé* ou *Not Configured*.

⁽²⁾ %HSC3 est disponible lorsque %HSC1 est configuré en tant que *Monophasé* ou *Not Configured*.

⁽³⁾ La fonction PTO est disponible sur les contrôleurs équipés de sorties transistor.

Mappage de sorties pour les fonctions expertes sur M221 Logic Controller

Les informations ci-dessous concernent les sorties transistor normales et rapides sur M221 Logic Controller :

Fonction		Sortie d'alarme	HSC	PLS / PWM / PTO / FREQGEN
Sortie rapide ⁽¹⁾	%Q0.0	X	–	<ul style="list-style-type: none"> • %PLS0 • %PWM0 • %PTO0 • %FREQGEN0
	%Q0.1	X	–	<ul style="list-style-type: none"> • %PLS1 • %PWM1 • %PTO⁽²⁾ • %FREQGEN1
Sortie normale ⁽³⁾ (selon la référence du contrôleur)	%Q0.2	X	Sortie réflexe 0 pour %HSC0 ou %HSC2	<ul style="list-style-type: none"> • %PTO⁽⁴⁾ • %FREQGEN2
	%Q0.3	X	Sortie réflexe 1 pour %HSC0 ou %HSC2	<ul style="list-style-type: none"> • %PTO⁽⁵⁾ • %FREQGEN3
	%Q0.4	X	Sortie réflexe 0 pour %HSC1 ou %HSC3	Direction %PTOx
	%Q0.5	X	Reflex output 1 for %HSC1 or %HSC3	Direction %PTOx
	%Q0.6	X	–	Direction %PTOx
	%Q0.7	X	–	Direction %PTOx
	%Q0.8	–	–	Direction %PTOx
	%Q0.9	–	–	Direction %PTOx
	%Q0.10	–	–	Direction %PTOx
	%Q0.11	–	–	Direction %PTOx
	%Q0.12	–	–	Direction %PTOx
	%Q0.13	–	–	Direction %PTOx
	%Q0.14	–	–	Direction %PTOx
	%Q0.15	–	–	Direction %PTOx

X Oui

- Non

(1) Les fonctions de sortie rapide ne sont disponibles que sur les contrôleurs équipés de sorties transistor.

(2) Direction %PTO0 en mode de sortie CW/CCW ou %PTO1 (non disponible si %PTO0 est configuré en mode de sortie CW/CCW) ou %PTOx dans les autres cas.

(3) %Q0.2 et %Q0.3 sont des sorties rapides sur les contrôleurs TM221C40U et TM221CE40U

(4) %PTO2 sur les contrôleurs TM221C40U et TM221CE40U ou direction %PTOx dans les autres cas.

(5) Direction %PTO2 en mode de sortie CW/CCW sur les contrôleurs TM221C40U et TM221CE40U ou %PTO3 (non disponible si %PTO2 est configuré en mode de sortie CW/CCW) sur les contrôleurs TM221C40U et TM221CE40U ou %PTOx dans les autres cas.

Informations générales sur la gestion des blocs fonction

Gestion des entrées de bloc fonction et des objets d'entrée

Les variables (entrées de bloc fonction et objets d'entrée) sont utilisées avec le front montant de l'entrée *Execute*. Pour modifier une variable, il est nécessaire de modifier les variables d'entrée et de déclencher à nouveau le bloc fonction. Cependant, certains blocs fonction proposent une option de mise à jour continue.

Gestion des sorties de bloc fonction et des objets de sortie

Les sorties *Done*, *Error*, *Busy* et *CmdAborted* s'excluent mutuellement : une seule d'entre elles peut être TRUE sur un bloc fonction. Si l'entrée *Execute* est TRUE, l'une de ces sorties est également TRUE.

Sur le front montant de l'entrée *Execute*, la sortie *Busy* est définie sur TRUE. Elle conserve la valeur TRUE lors de l'exécution du bloc fonction et elle est réinitialisée sur le front montant d'une des autres sorties (*Done*, *Error* et *CmdAborted*).

La sortie *Done* a la valeur TRUE lorsque l'exécution du bloc fonction a abouti.

Si une erreur est détectée, le bloc fonction s'arrête en définissant la sortie *Error* sur TRUE et le code d'erreur est contenu dans la sortie *ErrId*.

Les sorties *Done*, *Error* et *CmdAborted* sont définies sur TRUE ou FALSE avec le front descendant de l'entrée *Execute*, conformément aux conditions suivantes :

- définies pour un cycle de tâche si l'exécution du bloc fonction est terminée et que l'entrée *Execute* a la valeur FALSE, puis réinitialisées à leurs valeurs par défaut.
- conservent leur valeur si l'exécution du bloc fonction est terminée et que l'entrée *Execute* a la valeur TRUE.

Lorsqu'une instance d'un bloc fonction reçoit une nouvelle entrée *Execute* avant d'être terminée (série de commandes sur la même instance), le bloc fonction ne génère aucun retour, tel que *Done*, concernant l'action précédente. Toutefois, la nouvelle commande est lancée sur le bloc fonction (l'état est *Busy*).

Gestion des erreurs

Tous les blocs comportent deux sorties qui peuvent signaler une erreur détectée lors de l'exécution du bloc fonction :

- *Error* = Le front montant de cette sortie indique qu'une erreur a été détectée.
- *ErrID* = Code de l'erreur détectée.

Fonctions avancées des entrées expertes

Contenu de cette partie

Fast Counter (%FC)	23
High Speed Counter (%HSC)	28

Présentation

Cette section décrit les fonctions avancées des entrées expertes.

Fast Counter (%FC)

Contenu de ce chapitre

Description	23
Configuration	24
Exemple de programmation	26

Utilisation des blocs fonction Fast Counter

Ce chapitre fournit des descriptions et des instructions de programmation concernant l'utilisation des blocs fonction *Fast Counter*.

Description

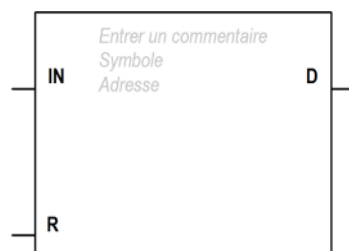
Introduction

Le bloc fonction *Fast Counter* ¹¹²³ sert de compteur croissant ou décroissant. Il peut compter le front montant d'entrées numériques jusqu'à 5 kHz de fréquence en mode de calcul mot simple ou mot double. Comme les blocs fonction *Fast Counter* sont gérés par des interruptions matérielles spécifiques, le maintien du taux d'échantillonnage maximal des fréquences peut varier en fonction de la configuration de votre application et de votre matériel.

Les blocs fonction *Fast Counter* %FC0, %FC1, %FC2 et %FC3 utilisent respectivement les entrées dédiées %I0.2, %I0.3, %I0.4 et %I0.5. Ces bits ne sont pas exclusivement réservés à ces blocs fonction. Leur affectation varie selon l'utilisation de ces ressources dédiées par d'autres blocs fonction.

Illustration

L'illustration suivante présente un bloc fonction *Fast Counter* en mode mot simple :



Entrées

Le bloc fonction *Fast Counter* a les entrées suivantes :

Libellé	Description	Valeur
IN	Activer	A l'état 1, la valeur est mise à jour selon les impulsions appliquées à l'entrée physique. A l'état 0, la valeur reste inchangée.
R	Réinitialiser (facultatif)	Utilisée pour initialiser le bloc. A l'état 1 : <ul style="list-style-type: none"> les valeurs %FC.P ou %FC.PD sont prises en compte. La valeur est remise à 0 lorsque le bloc est configuré en tant que compteur croissant ou définie sur %FC.P ou %FC.PD lorsqu'il est configuré en tant que compteur décroissant. Le bit Done (Terminé) %FC.D reprend sa valeur par défaut.

Sorties

Le bloc fonction *Fast Counter* comporte la sortie suivante :

Libellé	Description	Valeur
D	Terminé (%FCi.D)	Ce bit est à 1 dans les cas suivants : <ul style="list-style-type: none"> quand %FCi.V ou %FCi.VD atteint la valeur de présélection %FCi.P ou %FCi.PD pour un compteur croissant. quand %FCi.V ou %FCi.VD atteint 0 pour un compteur décroissant. Ce bit en lecture seule est remis à zéro lorsque %FCi.R est défini sur 1.

Configuration

Paramètres

Pour configurer les paramètres, suivez la procédure Configuration d'un bloc fonction (voir EcoStruxure Machine Expert - Basic, Guide de la bibliothèque des fonctions génériques) et lisez la description des Modes d'allocation de mémoire dans le document EcoStruxure Machine Expert - Basic - Guide d'utilisation (voir EcoStruxure Machine Expert - Basic, Guide d'utilisation).

Le bloc fonction *Fast Counter* a les paramètres suivants :

Paramètre	Description	Valeur
Utilisée	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	Adresse <i>%FCi Fast Counter</i>	Identificateur de l'instance, où i est compris entre 0 et le nombre d'objets disponibles sur ce contrôleur logique. Reportez-vous au tableau Nombre maximum d'objets (voir Modicon M221 - Logic Controller - Guide de programmation) pour connaître le nombre maximal de <i>Fast Counters</i> .
Entrée	<i>%IO.i</i>	Entrée dédiée associée à cette instance de bloc fonction. <i>%IO.2...%IO.5</i>
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, consultez le document EcoStruxure Machine Expert - Basic - Guide d'exploitation (Définition et utilisation des symboles).
Configuré	Indique si le comptage est croissant ou décroissant.	Les valeurs possibles sont : <ul style="list-style-type: none"> • Non utilisé • Compteur • Décompteur
Présélection	Valeur de présélection (<i>%FCi.P</i> ou <i>%FCi.PD</i>)	La valeur initiale peut être définie : <ul style="list-style-type: none"> • à l'aide de l'objet associé <i>%FCi.P</i>, de 1 à 65 535 en mode mot simple, • à l'aide de l'objet associé <i>%FCi.PD</i>, de 1 à 4 294 967 295 en mode mot double.
Double Word	Mode mot double	Si ce paramètre est sélectionné, utilisez le mode mot double. Sinon, utilisez le mode mot simple.
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Double-cliquez dans la colonne Commentaire et tapez le commentaire de votre choix.

Objets

Le bloc fonction *Fast Counter* est associé aux objets suivants :

Objet	Description	Valeur
<i>%FCi.V</i> <i>%FCi.VD</i>	Valeur actuelle	La valeur en cours augmente ou diminue selon la fonction sélectionnée (comptage croissant ou décroissant). Pour le comptage croissant, la valeur en cours est mise à jour et peut atteindre 65 535 en mode mot simple (<i>%FCi.V</i>) et 4 294 967 295 en mode mot double (<i>%FCi.VD</i>). Pour le comptage décroissant, la valeur en cours est la valeur de présélection <i>%FC.P</i> ou <i>%FC.PD</i> , et elle peut décroître jusqu'à 0.
<i>%FCi.P</i> <i>%FCi.PD</i>	Valeur de présélection	Une nouvelle valeur de présélection n'est prise en compte que si l'entrée R est active. Consultez la description dans le tableau Paramètres ci-dessus.
<i>%FCi.D</i>	Done	Consultez le tableau Sorties ci-dessus.

Opération

Ce tableau décrit les principales étapes des opérations du bloc fonction *Fast Counter* :

Opération	Action	Résultat
Comptage croissant	Un front montant apparaît au niveau de l'entrée de comptage croissant.	La valeur en cours %FCi.V est incrémentée de 1 unité.
	Lorsque la valeur de présélection %FCi.P ou %FCi.PD est atteinte.	Le bit de sortie Done %FCi.D passe à 1.
Comptage décroissant	Un front montant apparaît au niveau de l'entrée de comptage décroissant.	La valeur en cours %FCi.V est décrémentée de 1 unité.
	Lorsque la valeur est 0.	Le bit de sortie Done %FCi.D passe à 1.

Cas particuliers

Le tableau suivant contient une liste de cas spéciaux du bloc fonction *Fast Counter* :

Cas particulier	Description
Effet d'un redémarrage à froid (%S0=1)	Réinitialise les attributs de <i>Fast Counter</i> avec les valeurs configurées ou l'application utilisateur (voir Modicon M221 Logic Controller - Guide de programmation).
Effet d'un redémarrage à chaud (%S1=1)	Aucun effet (voir Modicon M221 Logic Controller - Guide de programmation).
Effet de l'arrêt du contrôleur	Le compteur <i>Fast Counter</i> arrête le comptage si le contrôleur est à l'état <i>STOPPED</i> ; il le reprend dès que le contrôleur revient à l'état <i>RUNNING</i> . Le comptage reprend à partir de la dernière valeur utilisée avant le passage à l'état <i>STOPPED</i> .

Exemple de programmation

Introduction

Dans cet exemple, l'application compte un nombre d'éléments jusqu'à 5000 tant que %I0.1 est défini sur 1. L'entrée %FC1 est l'entrée dédiée %I0.3. Lorsque la valeur de présélection est atteinte, %FC1.D est défini sur 1 et conserve la même valeur jusqu'à ce que %FC1.R soit commandé par le résultat de AND sur %I0.2 et %M0.

Programmation

Voici un exemple de bloc fonction *Fast Counter* :

Réseau	Instruction
0	BLK %FC1 LD %I0.1 IN LD %I0.2 AND %M0 R OUT_BLK LD D ST %Q0.0 END_BLK

NOTE: Pour obtenir le schéma à contacts équivalent, reportez-vous à la procédure de réversibilité (voir EcoStruxure Machine Expert - Basic, Guide de la bibliothèque des fonctions génériques).

High Speed Counter (%HSC)

Contenu de ce chapitre

Description	28
Compteur HSC et modes de comptage	32
Compteur HSC en mode fréquencemètre	38

Utilisation des blocs fonction High Speed Counter

Ce chapitre fournit des descriptions et des instructions de programmation concernant l'utilisation des blocs fonction *High Speed Counter*.

Description

Introduction

Le bloc fonction *High Speed Counter* **11123** peut être configuré par EcoStruxure Machine Expert - Basic pour exécuter l'une des fonctions suivantes :

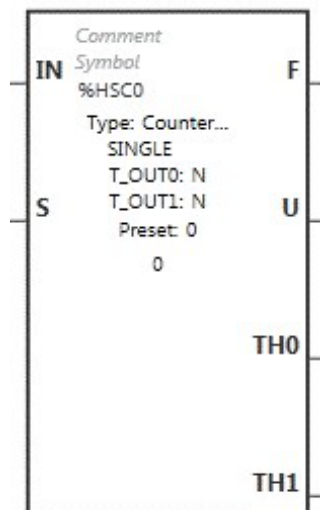
- Biphase [Impulsion / Direction]
- Biphase [Quadrature X1]
- Biphase [Quadrature X2]
- Biphase [Quadrature X4]
- Monophasé
- Fréquencemètre

Le bloc fonction *High Speed Counter* fonctionne à une fréquence maximale de 100 kHz quel que soit le mode de comptage, avec une plage de valeurs de 0 à 65535 en mode mot simple et de 0 à 4294967295 en mode mot double.

Le bloc fonction *High Speed Counter* utilise des entrées dédiées et des entrées/sorties auxiliaires. Pour plus d'informations sur les entrées et les sorties, reportez-vous au document M221 Logic Controller - Guide de référence du matériel.

Vous devez initialiser la fonction *High Speed Counter* dans l'onglet **Configuration** à l'aide de l'**Assistant HSC** avant d'utiliser une instance du bloc fonction. Reportez-vous à la section Configuration de compteurs HSC (voir Modicon M221 Logic Controller - Guide de programmation).

Représentation graphique



Entrées

Le bloc fonction *High Speed Counter* a les entrées suivantes :

Eti-quette	Description	Valeur
IN	Activer (requis) A l'état 1, la fonction de comptage ou la mesure de fréquence est activée. A l'état 0, la valeur en cours reste inchangée.	0 ou 1
S	Entrée de présélection. A l'état 1 : <ul style="list-style-type: none"> initialise la valeur avec la valeur de présélection pour : <ul style="list-style-type: none"> Biphasé [Quadrature X1], Biphasé [Quadrature X2], Biphasé [Quadrature X4] ou Biphasé [Impulsion / Direction] avec fonction de comptage décroissant en cours remet la valeur à 0 pour : <ul style="list-style-type: none"> Monophasé ou Biphasé [Impulsion / Direction] avec fonction de comptage croissant en cours <p>Cette fonction permet également d'initialiser la commande des sorties de seuil et prend en compte toutes les modifications apportées par un utilisateur aux valeurs de seuil définies dans la fenêtre de propriétés ou le programme.</p>	0 ou 1

Le bloc fonction *High Speed Counter* est associé aux objets d'entrée suivants :

Objet	Type	Description	Valeur
%HSCi.P %HSCi.PD	WORD DOUBLE WORD	Valeur de présélection	Consultez la section Entrées auxiliaires, page 33.
%HSCi.S0 %HSCi.S0D	WORD DOUBLE WORD	Seuil 0	Consultez la section Seuil de sortie en modes de comptage, page 33.
%HSCi.S1 %HSCi.S1D	WORD DOUBLE WORD	Seuil 1	Consultez la section Seuil de sortie en modes de comptage, page 33.
%HSCi.T	WORD	Base de temps	Consultez la section Bloc fonction %HSC en mode fréquencemètre, page 38.
%HSCi.R	BOOL	Activation de la sortie réflexe 0	A l'état 1, active la sortie réflexe 0.
%HSCi.S	BOOL	Activation de la sortie réflexe 1	A l'état 1, active la sortie réflexe 1.

NOTE: Les bits %HSCi.R et %HSCi.S activent ou désactivent respectivement les sorties réflexes, uniquement si le bloc fonction HSC est activé, c'est-à-dire, si %HSCi.IN est défini sur 1.

Sorties

Le bloc fonction *High Speed Counter* a les sorties suivantes :

Etiquette	Description	Valeur
F	Débordement Mis à 1 en cas de débordement arithmétique.	0 ou 1
U	Sens de comptage Réglé par le système, ce bit est utilisé par la fonction de comptage <i>Dual Phase</i> pour indiquer le sens de comptage.	0 : Comptage décroissant 1 : Comptage croissant
TH0	Bit de seuil 0 Mis à 1 lorsque la valeur en cours est supérieure ou égale à la valeur du seuil S0 (%HSCi.S0). Ne testez ce bit qu'une fois dans le programme, car il est mis à jour en temps réel. L'application utilisateur est responsable de la validité de la valeur au moment de son utilisation.	0 ou 1
TH1	Bit de seuil 1 Mis à 1 lorsque la valeur en cours est supérieure ou égale à la valeur de seuil S1 (%HSCi.S1). Ne testez ce bit qu'une fois dans le programme, car il est mis à jour en temps réel.	0 ou 1

Le bloc fonction *High Speed Counter* est associé aux objets de sortie suivants :

Objet	Type	Description	Valeur
%HSCi.V %HSCi.VD	WORD DOUBLE WORD	Valeur actuelle	Consultez les sections Compteur HSC en modes de comptage, page 32 et Bloc fonction %HSC en mode fréquencemètre, page 38. NOTE: La valeur actuelle peut être mise à jour indépendamment de l'appel du bloc fonction %HSC. %HSCi.V/%HSCi.VD peut être lu deux fois et donner des résultats différents dans le même cycle de tâche.
%HSCi.C %HSCi.CD	WORD DOUBLE WORD	Valeur de capture	Consultez la section Entrées auxiliaires, page 33.
%HSCi.U	BOOL	Sens de comptage	0 : Comptage décroissant 1 : Comptage croissant
%HSCi.F	BOOL	Débordement	0 : Pas de débordement 1 : Débordement de compteur

Propriétés

Le bloc fonction *High Speed Counter* a les propriétés suivantes :

Propriété	Valeur	Description
Utilisation	Case à cocher activée / désactivée	Indique si l'adresse est utilisée.
Adresse	%HSCi, où <i>i</i> est compris entre 0 et 3, selon le ou les types de compteurs configurés	<i>i</i> est l'identificateur de l'instance. Pour connaître le nombre maximum d'objets %HSC, consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller – Guide de programmation).
Symbole	Texte défini par l'utilisateur	Symbole identifiant cet objet de manière univoque. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert – Basic – Guide d'exploitation).
Présélection	<ul style="list-style-type: none"> De 0 à 65 535 pour %HSCi.P De 0 à 4 294 967 295 pour %HSCi.PD 	Valeur de présélection à utiliser pour initialiser la valeur en cours de HSC (%HSCi.P, %HSCi.PD). Non valide pour le Fréquencemètre.
S0	<ul style="list-style-type: none"> De 1 à 65 535 pour %HSCi.S0 De 1 à 4294967295 pour %HSCi.S0D 	La valeur de seuil 0 est utilisée pour comparaison avec la valeur en cours. La valeur de S0 doit être inférieure à S1 (%HSCi.S1).
S1	<ul style="list-style-type: none"> De 2 à 65 535 pour %HSCi.S1 De 2 à 4294967295 pour %HSCi.S1D 	La valeur de seuil 1 est utilisée pour comparaison avec la valeur en cours. La valeur de S1 doit être supérieure à S0 (%HSCi.S0).
Base de temps	100 ms ou 1 s pour %HSCi.T	Base de temps de la mesure de fréquence.
Commentaire	Texte défini par l'utilisateur	Commentaire à associer à cet objet.

Cas particuliers

Le tableau suivant répertorie les opérations spéciales du bloc fonction *High Speed Counter* :

Cas particulier	Description
Effet d'un redémarrage à froid (%S0=1)	Réinitialise les attributs du bloc fonction <i>High Speed Counter</i> sur les valeurs configurées par le programme.
Effet d'un redémarrage à chaud (%S1=1)	Aucun.
Effet de l'arrêt du contrôleur	Le bloc fonction <i>High Speed Counter</i> cesse de fonctionner et les sorties gardent leur état en cours. NOTE: A l'arrêt du contrôleur, les sorties réflexes sont définies sur 0 si Conserver les valeurs est sélectionné pour les sorties. Dans le cas où Conserver les valeurs n'est pas sélectionné, les sorties réflexes prennent leurs valeurs de repli. Pour plus d'informations sur la configuration du comportement de repli, reportez-vous à la section Comportement de repli (voir EcoStruxure Machine Expert - Basic - Guide d'exploitation).

Compteur HSC et modes de comptage

Introduction

Le bloc fonction *High Speed Counter* fonctionne à une fréquence maximale de 100 kHz quel que soit le mode de comptage, avec une plage de valeurs de 0 à 65 535 en mot simple et de 0 à 4 294 967 295 en mot double.

Les impulsions à compter sont appliquées de la manière suivante :

Fonction	Description	Type d'entrée	%HSC0	%HSC1	%HSC2	%HSC3
Biphasé [Impulsion / Direction]	Les impulsions sont appliquées à l'entrée physique associée à Entrée d'impulsion .	Entrée à impulsions (Pulse input)	%I0.0	%I0.6	–	–
	L'opération en cours (comptage croissant/décroissant) est définie par l'état de l' entrée Direction : <ul style="list-style-type: none"> 0 = comptage croissant 1 = comptage décroissant 	Entrée Direction	%I0.1	%I0.7	–	–
Biphasé [Quadrature X1], Biphasé [Quadrature X2] ou Biphasé [Quadrature X4]	Les 2 phases du codeur sont appliquées aux entrées physiques associées à Entrée d'impulsion – Phase A et Entrée d'impulsion – Phase B .	Entrée d'impulsion – Phase A	%I0.0	%I0.6	–	–
		Entrée d'impulsion – Phase B	%I0.1	%I0.7	–	–
Monophasé	Les impulsions sont appliquées à l'entrée physique associée à Entrée d'impulsion .	Entrée à impulsions (Pulse input)	%I0.0	%I0.6	%I0.1	%I0.7

NOTE: L'affectation des E/S est différente entre la plateforme Twido et la plage du contrôleur logique M221. Sur le contrôleur logique M221, la principale entrée d'impulsions est %I0.0 pour %HSC0 et %I0.6 pour %HSC1. Sur la plateforme Twido, la principale entrée d'impulsions est %I0.1 pour %HSC0 et %I0.7 pour %HSC1.

Seuils de sortie

Pendant le comptage, la valeur courante est comparée à deux seuils : $\%HSCi.S0$ ou $\%HSCi.S0D$ et $\%HSCi.S1$ ou $\%HSCi.S1D$.

Les modifications de ces valeurs de seuil sont prises en compte quelle que soit la valeur de l'entrée **Présélection**.

Les modifications des valeurs de seuil sont enregistrées dans le contrôleur logique (objets $\%HSCi.S0$, $\%HSCi.S1$, $\%HSCi.S0D$ et $\%HSCi.S1D$), mais non dans la fenêtre **Configuration** de EcoStruxure Machine Expert - Basic.

Selon le résultat des comparaisons, les objets bit $\%HSCi.TH0$ et $\%HSCi.TH1$ sont :

- mis à 1 si la valeur en cours est supérieure ou égale au seuil correspondant ;
- remis à 0 si la valeur en cours est inférieure au seuil correspondant.

Les sorties réflexes physiques peuvent être configurées pour répondre différemment dans les résultats de la comparaison des valeurs de seuil et de la valeur en cours des compteurs.

NOTE: Zéro, une ou deux sorties réflexes peuvent être configurées.

Pour plus d'informations sur la configuration des sorties réflexes, reportez-vous à la section Configuration de compteurs biphasés et monophasés (voir Modicon M221 Logic Controller - Guide de programmation).

$\%HSCi.U$ est une sortie de ce bloc fonction qui indique le sens du compteur associé (1 pour le sens croissant, 0 pour le sens décroissant).

Entrées auxiliaires

Les opérations de comptage sont effectuées sur le front montant des impulsions et uniquement si le bloc fonction de comptage est activé (entrée **IN** à l'état 1).

Deux entrées facultatives sont utilisées pour le mode de comptage : **Entrée de capture** et **Entrée de présélection** :

- Un front montant sur l'**Entrée de capture** permet de capturer la valeur en cours ($\%HSCi.V$ ou $\%HSCi.VD$) et de la stocker dans $\%HSCi.C$ ou $\%HSCi.CD$. Les entrées de capture (si elles existent) sont identifiées par $\%I0.3$ pour $\%HSC0$ et par $\%I0.4$ pour $\%HSC1$.
- Un front montant sur l'**Entrée de présélection** initialise la valeur $\%HSCi.V$ ou $\%HSCi.VD$ avec la valeur prédéfinie pour :
 - Biphase [Quadrature X1]
 - Biphase [Quadrature X2]
 - Biphase [Quadrature X4]
 - Biphase [Impulsion / Direction] avec fonction de comptage décroissant en cours

L'**Entrée de présélection** réinitialise la valeur à 0 pour :

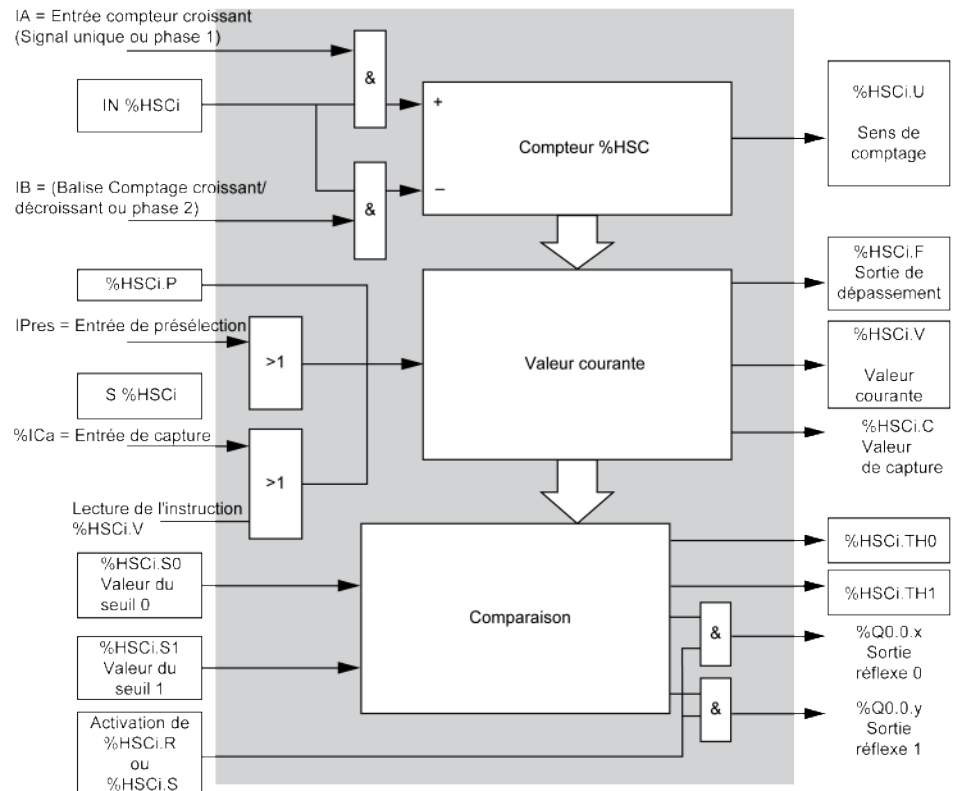
- Monophasé
- Biphase [Impulsion / Direction] avec fonction de comptage croissant en cours

Si l'**Entrée de présélection** auxiliaire est définie sur 1 avec l'entrée **IN** à 0 (la fonction est inhibée), les sorties ne sont pas surveillées et conservent leurs valeurs.

NOTE: $\%HSCi.F$ est également mis à 0. L'**Entrée de présélection** est identifiée par $\%I0.2$ pour $\%HSC0$ et/ou $\%I0.5$ pour $\%HSC1$.

Fonctionnement

Le diagramme décrit le fonctionnement du comptage en mode Mot simple (en mode Mot double, utilisez les variables de fonction Mot double) :



NOTE: Les sorties réflexes sont gérées de manière indépendante de la durée du cycle du contrôleur.

Chronogramme de Biphase [Impulsion / Direction]

Exemple de configuration des sorties réflexes :

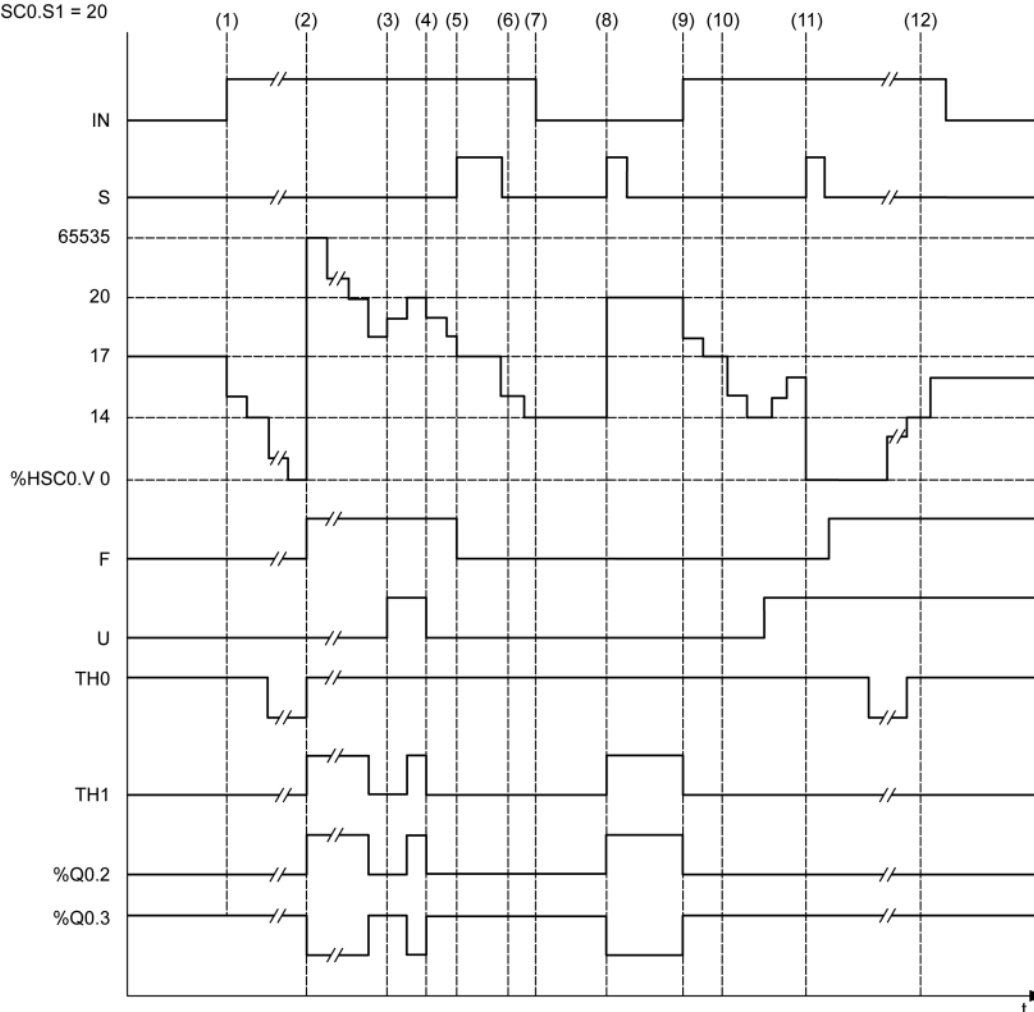
Sortie réflexe	Valeur < %HSC0.S0	%HSC0.S0 <= Valeur < %HSC0.S1	Valeur >= %HSC0.S1
%Q0.2	0	0	1
%Q0.3	1	1	0

Chronogramme :

%HSC0.P = 17

%HSC0.S0 = 14

%HSC0.S1 = 20



- (1) L'entrée IN est à 1, donc le mode de comptage décroissant démarre (%HSC0.U = 0, soit IB = 1)
- (2) La valeur courante atteint 0, donc l'indicateur de sortie F passe à 1 et %HSC0.V prend la valeur à 65535 au prochain comptage
- (3) Modification au niveau de l'entrée IB ; le compteur passe en mode croissant et %HSC0.U = 1.
- (4) L'entrée IB est mise à 1, donc le mode de comptage est décroissant et %HSC0.U est mis à 0.
- (5) L'entrée S est mise à 1 pendant le décomptage, donc %HSC0.V est initialisé à la valeur de présélection %HSC0.P = 17
- (6) S est remis à 0 et la valeur de présélection %HSC0.P passe à 20.
- (7) L'entrée IN est mise à 0, donc la fonction est inhibée ; %HSC0.V est maintenu
- (8) S est mis à 1, donc la nouvelle valeur de présélection (%HSC0.P = 20) est prise en compte et les sorties réflexes sont mises à jour **Remarque** : Si une entrée de présélection auxiliaire est utilisée à la place de S, les sorties réflexes ne sont pas actualisées conformément à la famille de contrôleurs Twido.
- (9) L'entrée IN est mise à 1 et la fonction redémarre en mode de comptage décroissant
- (10) La valeur de seuil %HSC0.S1 est définie sur 17.
- (11) L'entrée S activée permet d'affecter la nouvelle valeur de seuil S1 lors du prochain comptage et remet %HSC0.V à 0
- (12) Une capture de la valeur courante %HSC0.V est effectuée de sorte que %HSC0.C = 14

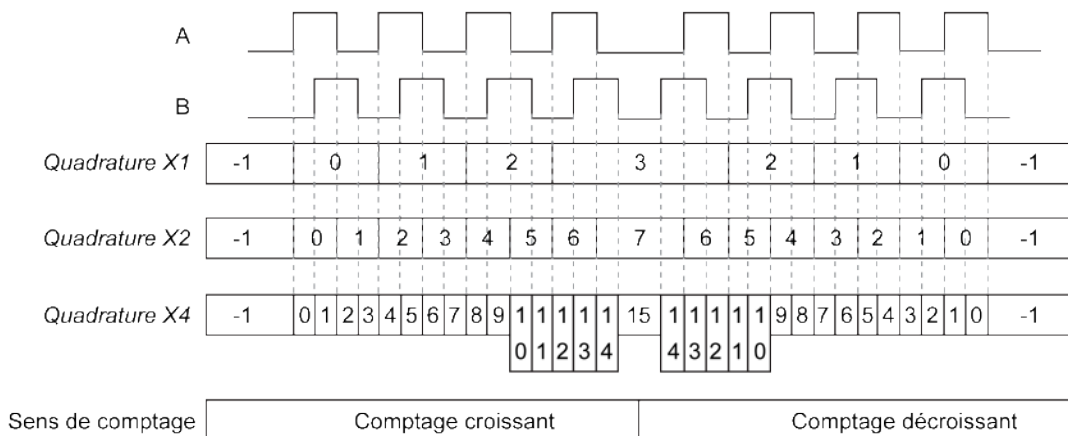
NOTE: Pour que les sorties réflexes configurées soient actives, %HSC0.R et %HSC0.S doivent être définis sur TRUE.

Chronogramme de Biphasé [Quadrature X1], Biphasé [Quadrature X2] et Biphasé [Quadrature X4]

Un codeur physique fournit deux signaux avec un décalage de 90°, qui permettent au compteur de compter les impulsions et de détecter le sens :

X1	1 impulsion par cycle de codeur
X2	2 impulsions par cycle de codeur
X4	4 impulsions par cycle de codeur

Chronogramme :



Quadrature X1 Lorsque la voie A devance la voie B, le compteur est incrémenté sur le front montant de la voie A. À l'inverse, lorsque la voie B devance la voie A, il est décrémenté sur le front descendant de la voie A.

Quadrature X2 Le compteur est incrémenté ou décrémenté sur chaque front de la voie A, selon la voie qui devance l'autre. Deux impulsions sont comptées ou décomptées à chaque cycle.

Quadrature X4 Le compteur est incrémenté ou décrémenté sur chaque front des voies A et B, selon la voie qui devance l'autre. Quatre impulsions sont comptées ou décomptées à chaque cycle.

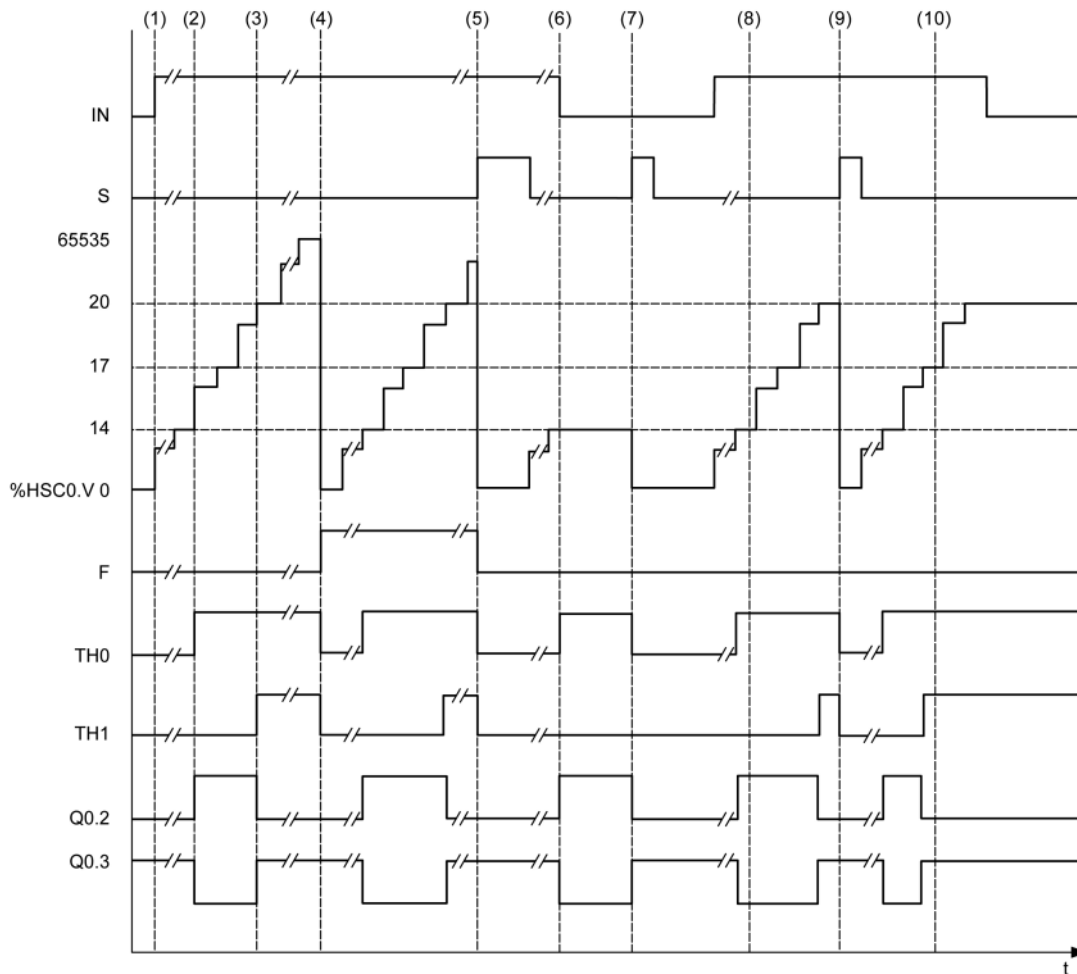
Chronogramme de Monophasé

Exemple de configuration des sorties réflexes :

Sortie réflexe	Valeur < %HSC0.S0	%HSC0.S0 <= Valeur < %HSC0.S1	Valeur >= %HSC0.S1
%Q0.2	0	1	0
%Q0.3	1	0	1

Chronogramme :

%HSC0.P = 17
 %HSC0.S0 = 14
 %HSC0.S1 = 20



- (1)** IN mis à 1 : la fonction de comptage est activée (%HSC0.U = 1 car %HSC0 est un compteur croissant)
- (2)** %Q0.2 (sortie réflexe) et TH0 sont mis à 1
- (3)** TH1 est réglé sur 1
- (4)** La valeur maximale est atteinte, donc au prochain comptage %HSC0.Vest remis à 0 et F est mis à 1
- (5)** S est mis à 1, la valeur courante %HSC0.V est mise à 0
- (6)** La fonction courante est inhibée tant que IN est à 0
- (7)** Pendant que la fonction est inhibée, S est mis à 1, donc la valeur en cours est remise à 0
- (8)** La valeur de seuil S1 passe à 17
- (9)** S est mis à 1, donc la nouvelle valeur de S1 sera affectée lors du prochain comptage
- (10)** L'entrée de capture est mise à 1, donc %HSC0.C = 17

Compteur HSC en mode fréquencesmètre

Introduction

Le mode fréquencesmètre d'un bloc fonction *High Speed Counter* permet de mesurer la fréquence d'un signal périodique en Hz sur l'entrée IA (entrée d'impulsion de phase A).

La plage de fréquences mesurables est comprise entre 1 Hz et 100 kHz, avec une plage de valeurs de 0 à 4 294 967 295 en mode Mot double.

Il est possible de choisir entre deux bases de temps, ce choix étant effectué par l'objet `%HSC.T` (base de temps) :

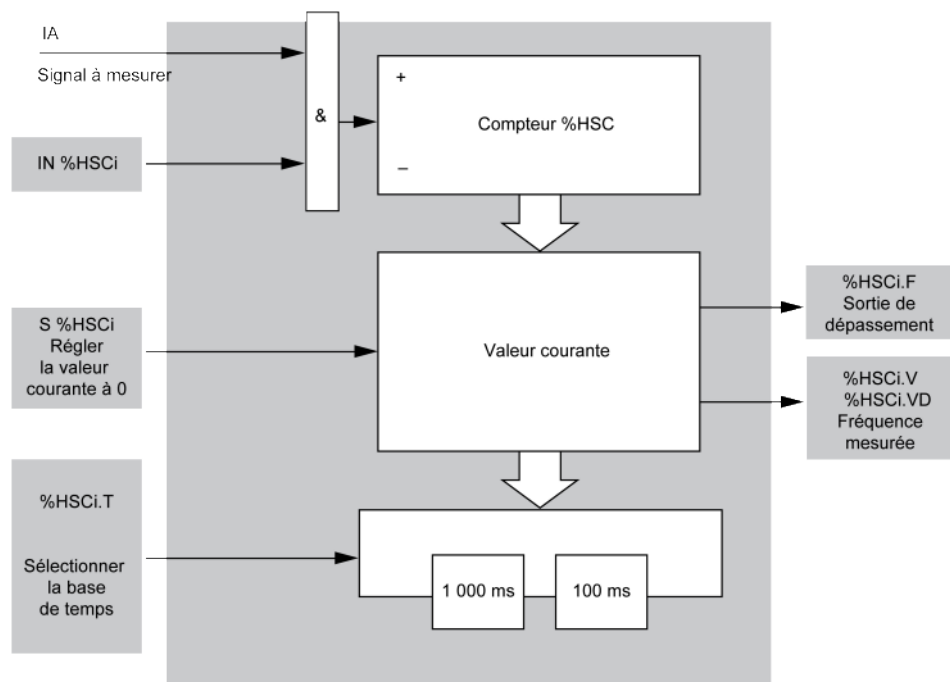
Base de temps	Précision	Mise à jour
100 ms	0,01 % pour 100 kHz 10% pour 100 Hz	10 fois par seconde
1 s	0,001% pour 100 kHz 10% pour 10 Hz	Une fois par seconde

Mesure de la précision

$$\text{Précision}(\%) = \frac{1}{f[\text{Hz}]} \times \frac{1}{TB[s]} \times 100$$

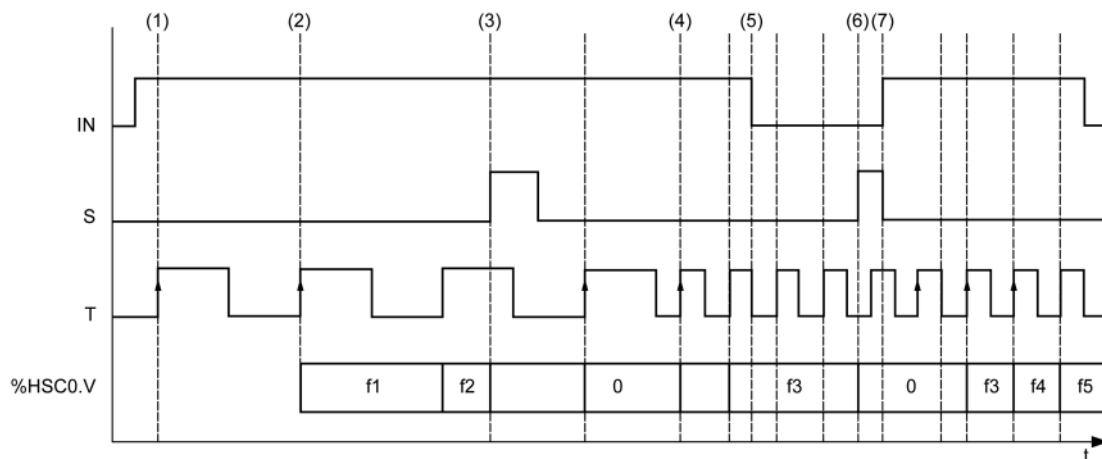
Fonctionnement

Cette illustration décrit le schéma de fonctionnement du mode fréquencesmètre :



Chronogramme

Le chronogramme suivant illustre l'utilisation d'un *High Speed Counter* en mode fréquencemètre :



- (1) La première mesure de fréquence débute sur un front montant du signal *TB*
- (2) *%HSC0.V* (ou *%HSC0.VD*) est mis à jour après une période de *TB*
- (3) Les entrées *IN* et *S* sont mises à 1, donc *%HSC0.V* (ou *%HSC0.VD*) est mis à 0
- (4) *%HSC0.T* est réglé sur 100 ms, donc la mesure est annulée et une nouvelle mesure commence
- (5) L'entrée *IN* est mise à 0, donc la fonction de mesure de fréquence est inhibée et *%HSC0.V* (ou *%HSC0.VD*) conserve sa valeur
- (6) *S* est mis à 1, donc la valeur *%HSC0.V* (ou *%HSC0.VD*) est mise à 0
- (7) *S* est mis à 0 et *IN* est mis à 1, donc la mesure commencera au prochain front montant du signal *TB*

Fonctions avancées des sorties expertes

Contenu de cette partie

Bloc fonction Pulse (%PLS)	41
Bloc fonction Pulse Width Modulation (%PWM)	48
Drive (%DRV)	53
Bloc fonction Pulse Train Output (%PTO).....	78
Générateur de fréquence (%FREQGEN)	149

Présentation

Cette section décrit les fonctions avancées des sorties expertes.

Bloc fonction Pulse (%PLS)

Contenu de ce chapitre


Description	41
Configurer des blocs de fonctions	42
Exemple de programmation	47

Utilisation des blocs fonction Pulse

Ce chapitre fournit des descriptions et des instructions de programmation concernant l'utilisation des blocs fonction *Pulse*.

Description

Introduction

Le bloc fonction *Pulse*  (impulsion) est utilisé pour générer des signaux carrés.

Deux blocs fonction *Pulse* sont disponibles sur la voie de sortie dédiée %Q0.0 ou %Q0.1. Les Logic Controller munis de sorties relais pour ces deux voies ne prennent pas en charge le bloc fonction *Pulse*. Pour plus d'informations sur les entrées et les sorties, reportez-vous au document M221 Logic Controller - Guide de référence du matériel.

Le bloc fonction *Pulse* n'autorise qu'une largeur de signal (cycle de service) de 50 %.

Vous pouvez choisir de limiter le nombre d'impulsions ou la période d'exécution du train d'impulsions. Cela peut être déterminé lors de la configuration et/ou mis à jour par le programme.

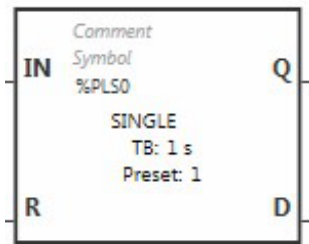
Vous devez configurer le bloc fonction *Pulse* dans la **Configuration > Générateurs d'impulsions** avant d'utiliser une instance du bloc fonction, reportez-vous à Configuration des générateurs d'impulsions (voir Modicon M221, Contrôleur logique, Guide de programmation).

Les caractéristiques de la fonction PLS sont présentées ci-après :

Caractéristique	Valeur
Nombre de voies	2
Fréquence minimale	1 Hz
Fréquence maximale	10000 Hz
Précision sur la fréquence	1 %

Illustration

Cette illustration décrit un bloc fonction *Pulse* :



Entrées

Le bloc fonction *Pulse* a les entrées suivantes :

Etiquette	Description	Valeur
IN	Activer	A l'état 1, l'impulsion est générée sur la voie de sortie dédiée. A l'état 0, la voie de sortie est paramétrée sur 0.
R	Remise à 0 (facultatif)	A l'état 1, les sorties %PLSi.Q et %PLSi.D sont mises à 0. Le nombre d'impulsions générées dans la période T est mis à 0.

Sorties

Le bloc fonction *Pulse* a les sorties suivantes :

Etiquette	Objet	Description	Valeur
Q	%PLSi.Q	Génération en cours	A l'état 1, indique que le signal <i>Pulse</i> est généré sur la voie de sortie dédiée configurée.
D	%PLSi.D	Génération terminée (facultatif)	A l'état 1, la génération du signal est terminée. Le nombre voulu d'impulsions a été généré.

Configurer des blocs de fonctions

Présentation

Pour configurer la ressource *Pulse Generator*, reportez-vous à la section Configuration de générateurs d'impulsions (voir Modicon M221 Logic Controller - Guide de programmation).

Pour configurer la ressource *Pulse Generator* en tant que PLS, reportez-vous à la section Configuration d'impulsion (voir Modicon M221 Logic Controller - Guide de programmation).

Paramètres

Le bloc fonction *Pulse* a les paramètres suivants :

Paramètre	Description	Valeur
Utilisée	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	%PLSi Adresse Pulse	Identificateur de l'instance, i étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets <i>Pulse</i> , consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert - Basic - Guide d'exploitation).
Présélection	Présélection de la période (%PLSi.P)	<ul style="list-style-type: none"> Base de temps = 1 s, %PLSi.P = 1 ou 2 Base de temps = 10 ms, 1 <= %PLSi.P <= 200 Base de temps = 1 ms, 1 <= %PLSi.P <= 2000 Base de temps = 0,1 ms, 1 <= %PLSi.P <= 20000
Num. Impulsion	Nombre d'impulsions (%PLSi.N, %PLSi.ND)	Pour produire un nombre illimité d'impulsions, affectez la valeur 0 à %PLS.N ou %PLS.ND.
Intensité	Sortie d'intensité (%PLSi.Q)	0 ou 1.
Done	Impulsion Done (%PLSi.D)	A l'état 1, la génération du signal est terminée. Le nombre voulu d'impulsions a été généré. Ce paramètre est ensuite remis à 0 en réinitialisant les entrées IN ou R sur 1.
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Double-cliquez dans la colonne Commentaire et tapez le commentaire de votre choix.

Objets

Le bloc fonction *Pulse* est associé aux objets suivants :

Objet	Description	Taille (bits)	Valeur par défaut	Plage	
%PLSi.P	Valeur de présélection	16	Présélection (définie dans Configuration > Générateurs d'impulsions)	Présélection %PLSi.P	Base de temps
				1 à 20000	0,1 ms
				1 à 2000	1 ms
				1 à 200	10 ms
			1 ou 2	1 s (par défaut)	
%PLSi.N	Nombre d'impulsions	16	0	0 à 32767	
%PLSi.ND		32	0	0 à 2147483647	

Règles d'utilisation

La période du signal de sortie *T* est réglée avec les paramètres **Présélection** et **Base de temps** de sorte que $T = \%PLSi.P \times \text{Base de temps}$.

Ce tableau indique les différentes périodes disponibles :

Base de temps	Fréquence
0,1 ms	0,5 à 10000 Hz
1 ms	0,5 à 1000 Hz
10 ms	0,5 à 100 Hz
1 s	0,5 à 1 Hz

La **Base de temps** est définie dans **Configuration > Générateurs d'impulsions** et n'est pas modifiable. Pour plus d'informations, reportez-vous à la section Configuration de générateurs d'impulsions (voir Modicon M221 Logic Controller - Guide de programmation).

Si **%PLSi.P** est :

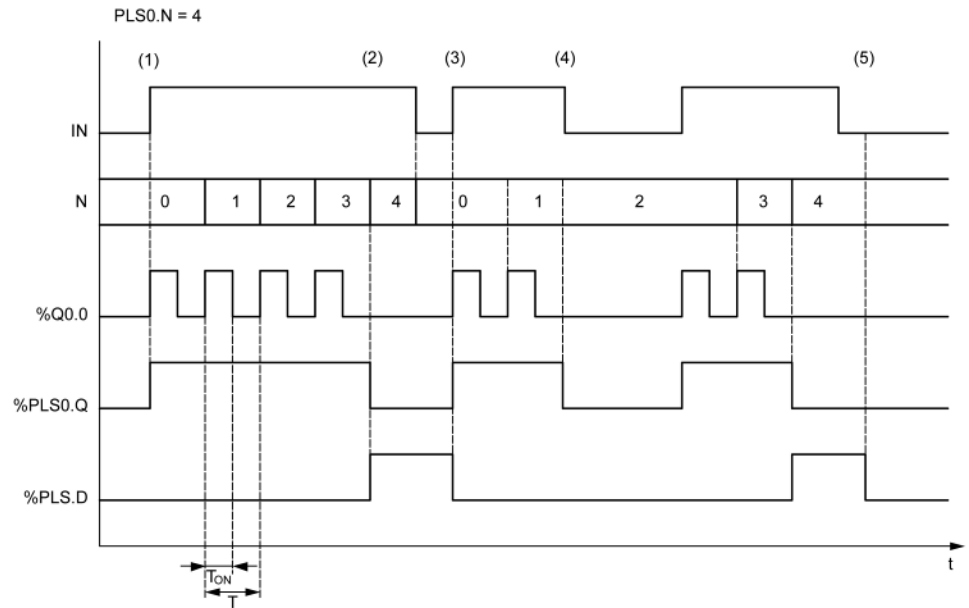
- modifié, la période du signal de sortie est modifiée à la fin de la période en cours.
- mis à 0, la fonction de génération d'impulsion est arrêtée.
- hors limites, le paramètre est forcé à 0 et la fonction de génération d'impulsion est arrêtée.

Si **%PLSi.N** (ou **%PLSi.ND** en mode **Mot double**) est :

- modifié, le nombre d'impulsions à générer est utilisé lors de la prochaine exécution de la fonction de génération d'impulsions (**%PLSi.D = 1** ou après que **%PLSi.R = 1**).
- mis à 0, un nombre illimité d'impulsions est généré.
- hors limites, le paramètre est forcé à 0.

Chronogramme

Le diagramme suivant indique la chronologie de fonctionnement du bloc fonction *Pulse* :



- (1) L'entrée *IN* est mise à 1, le signal d'impulsion est généré à la sortie dédiée (*%Q0.0*), donc *%PLSi.Q* est à 1
- (2) Le nombre d'impulsions atteint *%PLS0.N* (=4), donc la sortie d'indicateur Done (*%PLS0.D*) est définie sur 1 et la génération d'impulsions est arrêtée (*%PLS0.Q* = 0).
- (3) L'entrée *IN* est mise à 1, donc *%PLS0.D* est remis à 0.
- (4) L'entrée *IN* est mise à 0, donc la voie de sortie est mise à 0 et *%PLS0.Q* = 0 indique que la génération de signaux n'est pas active
- (5) *%PLS0.D* est mis à 0 par le réglage de l'entrée *R* sur 1

Cas particuliers

Cas particulier	Description
Effet d'un redémarrage à froid (%S0=TRUE)	<ul style="list-style-type: none"> La génération d'impulsions est arrêtée. Pendant l'initialisation du contrôleur, la sortie est remise à 0. Si, après l'initialisation du contrôleur : <ul style="list-style-type: none"> le contrôleur passe à l'état <i>STOPPED</i>, la stratégie de repli configurée est appliquée à la sortie. le contrôleur passe à l'état <i>RUNNING</i>, les paramètres de configuration sont restaurés.
Effet d'un redémarrage à chaud (%S1=TRUE)	<ul style="list-style-type: none"> La génération d'impulsions est arrêtée. Pendant l'initialisation du contrôleur, la sortie est remise à 0. Si, après l'initialisation du contrôleur : <ul style="list-style-type: none"> le contrôleur passe à l'état <i>STOPPED</i>, la stratégie de repli configurée est appliquée à la sortie. le contrôleur passe à l'état <i>RUNNING</i>, les paramètres de configuration sont restaurés ; toutefois, le nombre d'impulsions potentiellement déjà envoyées est remis à 0.⁽¹⁾
Effet à l'arrêt du contrôleur	<ul style="list-style-type: none"> La génération d'impulsions est arrêtée. Le comportement de repli dépend de la stratégie de repli configurée : <ul style="list-style-type: none"> Conserver les valeurs : les sorties sont réinitialisées à 0. Valeur de repli : les sorties sont définies sur les valeurs de repli configurées (voir EcoStruxure Machine Expert - Basic - Guide d'exploitation).
Effet d'une modification en ligne	Aucun(e)
Effet d'un court-circuit ou d'une surintensité sur une sortie dont l'adressage est effectué par le bloc fonction <i>Pulse</i> lors de la génération d'un nombre limité d'impulsions	<ul style="list-style-type: none"> La génération d'impulsions est arrêtée. Après suppression du court-circuit ou de la surintensité, la génération d'impulsions est relancée à partir de là où elle s'est arrêtée.
<p>⁽¹⁾ Si une instruction de sortie d'impulsion est en cours au moment du redémarrage à chaud, la génération d'impulsions lors du redémarrage du contrôleur ne prend pas en compte le nombre d'impulsions envoyées avant le redémarrage à chaud.</p>	

AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

- Évitez d'exécuter une commande de redémarrage à chaud (%S1=TRUE) lorsqu'une commande PLS est en cours.
- Si le redémarrage à chaud est incontournable, vous devez tenir compte de toutes les impulsions envoyées avant ce redémarrage.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Exemple de programmation

Introduction

Le bloc fonction *Pulse* peut être configuré comme dans cet exemple de programmation.

Programmation

Voici un exemple de bloc fonction *Pulse* :

Réseau	Instruction
0	BLK %PLS0 LD %M1 IN LD %M0 R OUT_BLK LD Q ST %Q0.5 LD D ST %M10 END_BLK

NOTE: Pour obtenir le schéma à contacts équivalent, reportez-vous à la procédure de réversibilité (voir EcoStruxure Machine Expert - Basic, Guide de la bibliothèque des fonctions génériques).

Bloc fonction Pulse Width Modulation (%PWM)

Contenu de ce chapitre


Description	48
Configuration de bloc fonction	49
Exemple de programmation	52

Utilisation des blocs fonction Pulse Width Modulation

Ce chapitre fournit des descriptions et des instructions de programmation concernant l'utilisation des blocs fonction *Pulse Width Modulation*.

Description

Introduction

Le bloc fonction *Pulse Width Modulation*  génère un signal d'onde variable sur une voie de sortie dédiée, %Q0.0 ou %Q0.1, avec une largeur (et donc un cycle de service) variable.

Les contrôleurs équipés de sorties relais pour ces deux voies ne prennent pas en charge cette fonction.

%PWM0 utilise la sortie dédiée %Q0.0 et %PMW1, la sortie dédiée %Q0.1. Il est également possible de configurer les blocs fonction %PLS pour utiliser ces mêmes sorties dédiées. Vous pouvez configurer l'une ou l'autre de ces fonctions (mais pas les deux) pour une sortie dédiée donnée.

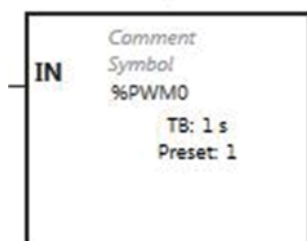
Vous devez configurer le bloc fonction *Pulse Width Modulation* dans **Configuration > Générateurs d'impulsions** avant d'en utiliser une instance. Consultez Configuration de générateurs d'impulsions (voir Modicon M221 Logic Controller - Guide de programmation).

Les caractéristiques de la fonction PWM sont présentées ci-après :

Caractéristique	Valeur
Nombre de voies	2
Fréquence minimale	1 Hz
Fréquence maximale	10000 Hz
Précision sur la fréquence	1 %

Illustration

L'illustration suivante représente le bloc fonction *Pulse Width Modulation* :



Entrées

Le bloc fonction *Pulse Width Modulation* comporte l'entrée suivante :

Libellé	Objet	Description	Valeur
IN	%PWMi.IN	Activer	A l'état 1, le signal <i>Pulse Width Modulation</i> est généré sur la voie de sortie. A l'état 0, la voie de sortie est paramétrée sur 0.

Configuration de bloc fonction

Présentation

Pour configurer la ressource *Pulse Generator*, reportez-vous à la section Configuration de générateurs d'impulsions (voir Modicon M221 Logic Controller - Guide de programmation).

Pour configurer la ressource *Pulse Generator* en tant que PWM, reportez-vous à la section Configuration de la modulation de largeur d'impulsion (voir Modicon M221 Logic Controller - Guide de programmation).

Propriétés

Le bloc fonction *Pulse Width Modulation* a les propriétés suivantes :

Propriété	Valeur	Description
Utilisée	Case à cocher activée / désactivée	Indique si l'adresse est utilisée.
Adresse	%PWMi où <i>i</i> est 0 ou 1	<i>i</i> est l'identificateur de l'instance. Pour connaître le nombre maximum d'objets <i>PWM</i> , consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Texte défini par l'utilisateur	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert - Basic - Guide d'exploitation).
Présélection	<ul style="list-style-type: none"> • %PWMi.P = 1 si Base de temps = 1 s • $1 \leq \%PWMi.P \leq 100$ si Base de temps = 10 ms • $1 \leq \%PWMi.P \leq 1000$ si Base de temps = 1 ms • $1 \leq \%PWMi.P \leq 10000$ si Base de temps = 0,1 ms 	Présélection de la période
Cycle de service	De 0 à 100 NOTE: les valeurs supérieures à 100 sont considérées comme égales à 100.	Le Cycle de service est contrôlé par l'objet %PWMi.R et correspond au pourcentage du signal à l'état 1 pendant la période. La largeur de l'état 1 (<i>Tp</i>) est donc égale à : $TP = T \times (\%PWMi.R/100)$. L'application utilisateur écrit la valeur de %PWMi.R.
Commentaire	Texte défini par l'utilisateur	Commentaire à associer à cet objet.

NOTE: les propriétés **Nombre d'impulsions**, **Courant** et **Terminé** qui figurent dans le tableau **Propriétés du générateur d'impulsions** sous l'onglet **Programmation** ne s'appliquent pas à la fonction PWM.

Objets

Le bloc fonction *Pulse Width Modulation* est associé aux objets suivants :

Objet	Description	Taille (bits)	Valeur par défaut	Plage	
%PwMi.P	Valeur de présélection	16	Présélection (définie dans Configuration > Générateurs d'impulsions)	Présélection % PwMi.P	Base de temps
				1 à 10000	0,1 ms
				1 à 1000	1 ms
				1 à 100	10 ms
				1	1 s (par défaut)
%PwMi.R	Cycle de service (<i>Ratio</i>)	16	0	0 à 100	

Si %PwMi.P est :

- modifié, la période du signal de sortie est modifiée à la fin de la période en cours.
- mis à 0, la fonction de génération d'impulsion est arrêtée.
- hors limites, le paramètre est forcé à 0 et la fonction de génération d'impulsion est arrêtée.

Si %PwMi.R est :

- mis à 0, la fonction de génération d'impulsions est arrêtée (sortie mise à 0).
- défini sur 100, le signal de sortie est mis à 1.
- modifié, le ratio du signal de sortie est modifié à la fin de la période en cours.
- hors limites, le paramètre est forcé à 0.

Base de temps

La **base de temps** est définie dans **Configuration > Générateurs d'impulsions** et n'est modifiable que dans l'onglet **Configuration**. Pour plus d'informations, reportez-vous à la section Configuration de générateurs d'impulsions (voir Modicon M221 Logic Controller - Guide de programmation).

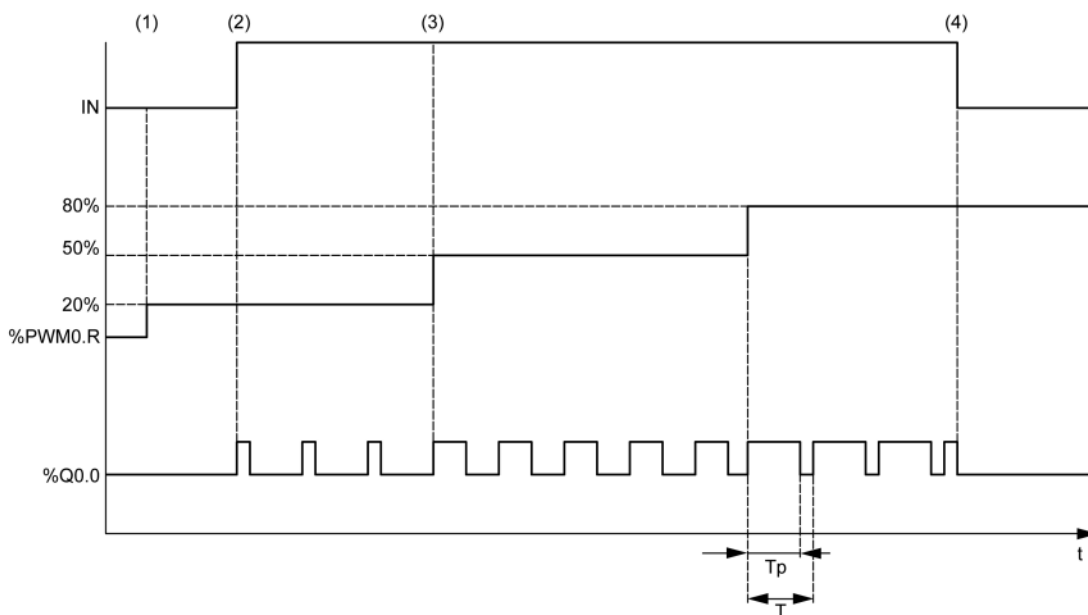
La période du signal de sortie T est réglée avec les paramètres **Présélection** et **Base de temps** de sorte que $T = \%PwMi.P \times \text{Base de temps}$.

Ce tableau indique les différentes périodes disponibles :

Base de temps	Plage de fréquence
0,1 ms	1 à 10000 Hz
1 ms	1 à 1000 Hz
10 ms	1 à 100 Hz
1 s	1 à 1 Hz

Chronogramme

Le diagramme suivant indique la chronologie de fonctionnement du bloc fonction *Pulse Width Modulation* :



- (1) Le ratio PWM ($\%PWMi.R$) est à 20 %, $IN = 0$, donc la génération d'impulsions n'est pas active
- (2) IN est mis à 1, donc la sortie PWM est activée
- (3) La largeur programmable (Tp) change avec $\%PWM.R$
- (4) IN est mis à 0, donc la fonction PWM est inhibée

Cas particuliers

Cas particulier	Description
Effet d'un redémarrage à froid ($\%S0=TRUE$)	<ul style="list-style-type: none"> • La génération d'impulsions est arrêtée. • Pendant l'initialisation du contrôleur, la sortie est remise à 0. • Si, après l'initialisation du contrôleur : <ul style="list-style-type: none"> ◦ le contrôleur passe à l'état <i>STOPPED</i>, la stratégie de repli configurée est appliquée à la sortie. ◦ le contrôleur passe à l'état <i>RUNNING</i>, les paramètres de configuration sont restaurés.
Effet d'un redémarrage à chaud ($\%S1=TRUE$)	<ul style="list-style-type: none"> • La génération d'impulsions est arrêtée. • Pendant l'initialisation du contrôleur, la sortie est remise à 0. • Si, après l'initialisation du contrôleur, ce dernier passe à l'état <i>STOPPED</i>, la stratégie de repli configurée est appliquée à la sortie.
Effet à l'arrêt du contrôleur	<ul style="list-style-type: none"> • La génération d'impulsions est arrêtée. • Le comportement de repli dépend de la stratégie de repli configurée : <ul style="list-style-type: none"> ◦ Conserver les valeurs : les sorties sont réinitialisées à 0. ◦ Valeur de repli : les sorties sont définies sur les valeurs de repli configurées (voir EcoStruxure Machine Expert - Basic - Guide d'exploitation).
Effet d'une modification en ligne	Aucun(e)

Exemple de programmation

Introduction

Le bloc fonction *Pulse Width Modulation* peut être configuré comme dans cet exemple de programmation.

Exemple de programmation

Dans cet exemple :

- La largeur du signal est modifiée par le programme en fonction de l'état des entrées %I0.0 et %I0.1 du contrôleur.
- La base de temps est de 10 ms.
- La valeur prédéfinie %PWM0.P est de 50, donc le pas proportionnel est égal à 2 %.
- La période configurable T est égale à 500 ms.

Résultat :

- Si %I0.0 et %I0.1 sont définis sur 0, le ratio %PWM0.R est défini sur 20 % et la durée du signal à l'état 1 est alors égale à : 20 % x 500 ms = 100 ms.
- Si %I0.0 est à 1 et %I0.1 à 0, le ratio %PWM0.R est de 50 % (durée = 250 ms).
- Si %I0.0 et %I0.1 sont à 1, le ratio %PWM0.R est de 80 % (durée = 400 ms).

Exemples d'instructions *Pulse Width Modulation* :

Réseau	Instruction
0	LDN %I0.0 ANDN %I0.1 [%PWM0.R:=20]
1	LD %I0.0 ANDN %I0.1 [%PWM0.R:=50]
2	LD %I0.0 AND %I0.1 [%PWM0.R:=80]
3	BLK %PWM0 LD %I0.2 IN END_BLK

NOTE: Pour obtenir le schéma à contacts équivalent, reportez-vous à la procédure de réversibilité (voir EcoStruxure Machine Expert - Basic, Guide de la bibliothèque des fonctions génériques).

Drive (%DRV)

Contenu de ce chapitre

Description	53
Etats du variateur et du contrôleur logique.....	55
Ajout d'un bloc fonction Drive	57
Configurer des blocs de fonctions	58
MC_Power_ATV : Activer/désactiver l'étage de puissance.....	58
MC_Jog_ATV : Démarrer le mode Jog	60
MC_MoveVel_ATV : Déplacement à une vitesse spécifiée.....	63
MC_Stop_ATV : Arrêt de mouvement.....	66
MC_ReadStatus_ATV : Lire l'état de l'équipement	68
MC_ReadMotionState_ATV : Lire l'état de mouvement.....	70
MC_Reset_ATV : Acquiescement et réinitialisation d'erreur	72
Codes d'erreur	75

Description

Présentation

Les blocs fonction Drive  permettent le contrôle de variateurs (tels que les variateurs de vitesse Altivar) par un contrôleur M221 Logic Controller. Exemple :

- contrôle de la vitesse d'un moteur géré par un variateur ATV et mise à jour en continu ;
- surveillance de l'état du variateur ATV et du moteur ;
- gestion des erreurs détectées dans le variateur ATV.

Les communications sont établies par l'intermédiaire de l'une des méthodes suivantes :

- Configuration de l'une des lignes série du Logic Controller en tant que scrutateur d'E/S Modbus série (voir Modicon M221 Logic Controller - Guide de programmation) à l'aide du protocole Modbus RTU.
- Configuration du port Ethernet en tant que Modbus TCP IOScanner.

Dans EcoStruxure Machine Expert - Basic, vous devez d'abord ajouter les types de variateurs ATV ciblés au scrutateur d'E/S Modbus Série ou au Modbus TCP IOScanner. Cela permet de configurer des voies prédéfinies et des demandes d'initialisation autorisant la lecture et l'écriture de données dans des registres spécifiques sur le variateur ATV, notamment par exemple :

- **ETA** Mot d'état
- **ETI** Mot d'état étendu
- **RFRD** Vitesse en sortie (tr/min)
- **DPO** Code de la dernière erreur
- **CMD** Mot de contrôle

Les données sont transférées par le biais du type de requête Modbus **FC23 - Lecture/écriture de plusieurs registres**. Il est ainsi possible pour le programme de lire les données des registres **ETA**, **ETI** et **DPO** et en même temps d'écrire dans le registre **CMD** à l'aide d'une seule requête Modbus.

Les blocs fonction Drive monoaxes sont disponibles dans l'onglet **Programmation** de EcoStruxure Machine Expert - Basic :

Bloc fonction	Description
MC_Power_ATV, page 58	Ce bloc fonction active ou désactive la phase d'alimentation d'un équipement.
MC_Jog_ATV, page 60	Ce bloc fonction démarre le mode de fonctionnement Jog sur un équipement.
MC_MoveVel_ATV, page 63	Ce bloc fonction indique une vitesse cible pour un équipement.
MC_Stop_ATV, page 66	Ce bloc fonction arrête le mouvement en cours sur un équipement.
MC_ReadStatus_ATV, page 68	Ce bloc fonction renvoie des informations d'état concernant un équipement.
MC_ReadMotionState_ATV, page 70	Ce bloc fonction renvoie des informations d'état concernant le mouvement en cours sur un équipement.
MC_Reset_ATV, page 72	Ce bloc fonction permet de réinitialiser les erreurs d'équipement relatives à l'état du variateur, page 55 et d'acquiescer les erreurs MC_Power_ATV, page 58.

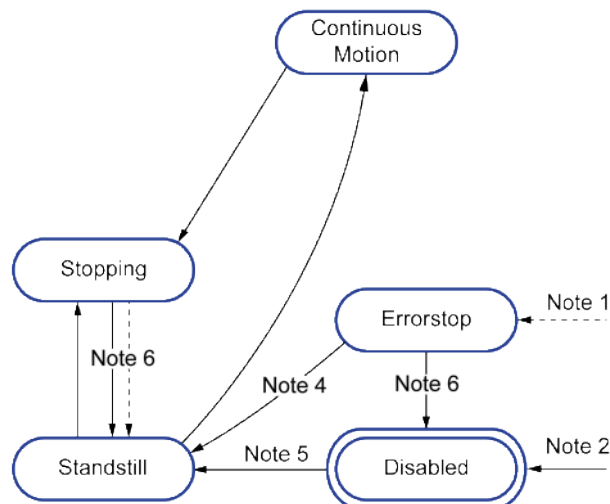
Un maximum de 16 instances de chaque bloc fonction Drive peuvent être utilisées dans un programme à un moment donné.

Lors de l'ajout d'un équipement au scrutateur d'E/S Modbus Série ou au Modbus TCP IOScanner, EcoStruxure Machine Expert - Basic utilise un objet *%DRVn* pour attribuer un axe à cet équipement, où *n* est le numéro du variateur ATV. Chaque fois que vous ajoutez un bloc fonction Drive à votre programme, vous devez l'associer à un axe, en créant un lien entre le bloc fonction, l'axe et l'équipement cible dans le scrutateur d'E/S Modbus Série ou le Modbus TCP IOScanner.

Etats du variateur et du contrôleur logique

Schéma des états du variateur

Le variateur est toujours dans un des états définis dans le schéma ci-dessous. Si un bloc fonction Drive est exécuté ou si une erreur se produit, cela peut engendrer une transition d'état :



Remarque 1 Depuis n'importe quel état si une erreur se produit.

Remarque 2 Depuis n'importe quel état (en l'absence de *ErrorAxis*) lorsque *MC_Power_ATV.status* est à 0.

Remarque 3 Transition de l'état *ErrorStop* à *Disabled* uniquement si *%MC_Reset_ATV.Done* = 1 et *%MC_Power_ATV.status* = 0.

Remarque 4 Transition de l'état *ErrorStop* à *Standstill* uniquement si *%MC_Reset_ATV.Done* = 1 et *%MC_Power_ATV.Enable* = 1 et *%MC_Power_ATV.Status* = 1.

Remarque 5 Transition de l'état *DISABLED* à *Standstill* uniquement si *%MC_Power_ATV.Enable* = 1 et *%MC_Power_ATV.Status* = 1.

Remarque 6 Transition de l'état *Stopping* à *Standstill* uniquement si *%MC_Stop_ATV.Done* = 1 et *%MC_Stop_ATV.Execute* = 0.

Ce tableau décrit les états du variateur :

Etat	Description
<i>Désactivé</i>	Etat initial. Le variateur n'est pas dans un état de fonctionnement ou dans un état d'erreur.
<i>Immobile</i>	Le variateur est dans un état de fonctionnement (ETA = 16#xx37) et <i>Velocity</i> = 0 (RFRD = 0).
<i>ErrorStop</i>	Le variateur est dans un état d'erreur (ETA = 16#xxx8).
<i>Mouvement continu</i>	Le variateur est dans un état de fonctionnement (ETA = 16#xx37) et <i>Velocity</i> ≠ 0 (RFRD ≠ 0).
<i>Arrêt</i>	Le bloc fonction <i>MC_Stop_ATV</i> est en cours d'exécution.

Il est possible d'utiliser le bloc fonction *MC_ReadStatus_ATV*, page 68 pour lire l'état du variateur ATV.

Changements d'états du contrôleur logique

Le tableau suivant explique comment les blocs fonction Drive sont influencés par les changements d'état du contrôleur logique :

Etat du contrôleur logique	Impact sur les blocs fonction Drive
<i>RUNNING</i>	Les blocs fonction Drive sont exécutés normalement selon la logique utilisateur.
<i>STOPPED</i>	<p>Les axes de variateur configurés sont arrêtés lorsque le contrôleur passe à l'état <i>STOPPED</i>, à moins que l'option Comportement de repli ne soit paramétrée pour conserver les valeurs.</p> <p>Si l'option Comportement de repli est paramétrée sur Valeurs de repli, la commande 0x00 est envoyée au variateur ATV, ce qui produit un état Switch on Disabled (NST). Autrement, si l'option Comportement de repli est paramétrée sur Conserver les valeurs, aucune mesure n'est prise (la commande n'est pas changée).</p>
<i>HALTED</i>	<p>Les axes de variateur configurés sont arrêtés lorsque le contrôleur passe à l'état <i>HALTED</i>, à moins que l'option Comportement de repli ne soit paramétrée pour conserver les valeurs.</p> <p>Si l'option Comportement de repli est paramétrée sur Valeurs de repli, la commande 0x00 est envoyée au variateur ATV, ce qui produit un état Switch on Disabled (NST). Dans le cas contraire, si l'option Comportement de repli est paramétrée sur Conserver les valeurs, aucune mesure n'est prise (la commande n'est pas changée).</p>
<i>POWERLESS, EMPTY</i>	<p>Les blocs fonction Drive ne sont pas exécutés (le scrutateur d'E/S Modbus Série ou le Modbus TCP IOScanner est arrêté).</p> <p>Cela est également le cas lorsque l'application du contrôleur est mise à jour.</p>

NOTE: Lorsque l'état du contrôleur est *HALTED* ou *STOPPED* et que vous avez sélectionné **Conserver les valeurs**, le variateur ne reçoit plus de commandes du contrôleur. Par conséquent, c'est au variateur de déterminer l'état approprié à prendre. Si vous avez sélectionné **Conserver les valeurs** propres au variateur, vous devez prendre ce paramètre en compte dans votre analyse des risques et des dangers car ce comportement peut induire des conséquences possiblement critiques.

▲ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

Assurez-vous qu'une évaluation des risques est effectuée et respectée conformément à la norme EN/ISO 12100 pendant la conception de votre machine.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Ajout d'un bloc fonction Drive

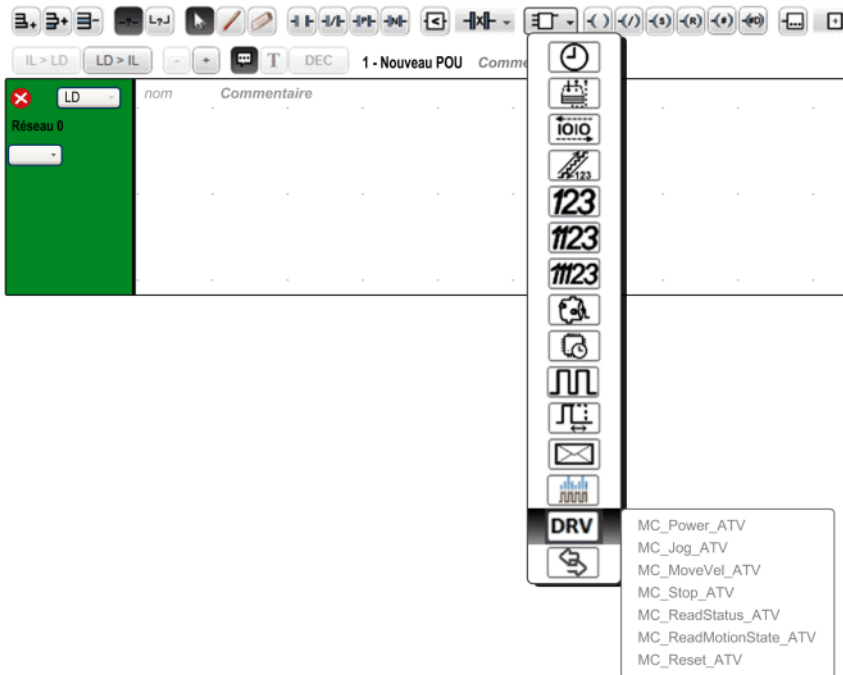
Prérequis

Conditions requises pour ajouter un bloc fonction Drive :

- Un scrutateur d'E/S Modbus Série ou un Modbus TCP IOScanner doit être configuré sur une ligne série ou Ethernet.
- Les variateurs ATV à contrôler doivent être ajoutés et configurés (voir Modicon M221 Logic Controller - Guide de programmation) sur le scrutateur d'E/S Modbus série ou Modbus TCP IOScanner.

Ajout d'un bloc fonction Drive

Pour ajouter une instance d'un bloc fonction Drive, procédez comme suit :

Éta-pe	Action
1	Sélectionnez l'onglet Programmation .
2	Sélectionnez Blocs Fonction > Drive , comme illustré dans la figure suivante : 
3	Cliquez sur le réseau pour y placer le bloc fonction sélectionné.
4	Associez les entrées / sorties du bloc fonction.

Suppression d'un bloc fonction

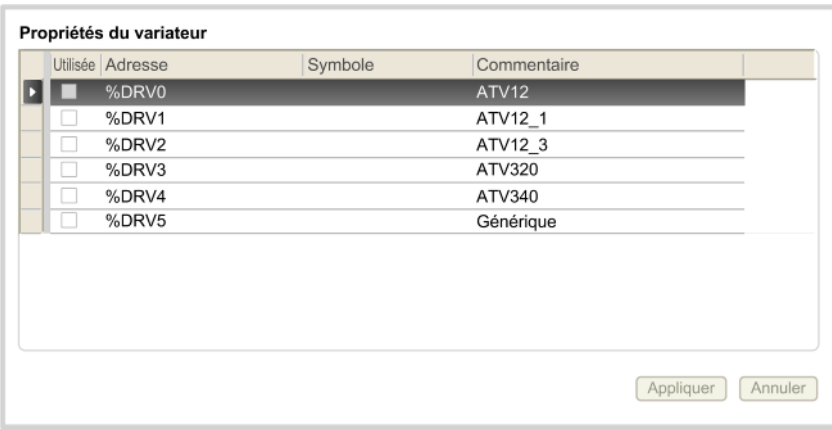
Pour supprimer une instance d'un bloc fonction Drive, procédez comme suit :

Éta-pe	Action
1	Dans l'onglet Programmation , cliquez sur l'instance du bloc fonction.
2	Cliquez sur Supprimer pour supprimer le bloc fonction sélectionné.

Configurer des blocs de fonctions

Configuration d'objets Drive

Chaque bloc fonction Drive est associé à un objet Drive (%DRV). Pour afficher une liste des objets Drive configurés, procéder comme suit :

Étape	Action
1	<p>Sélectionnez l'onglet Programmation > Outils, puis cliquez sur Objets Drive > Drive pour afficher les propriétés des objets Drive.</p> 
2	Actualisez les propriétés et cliquez sur Appliquer

Le bloc fonction Drive a les propriétés suivantes :

Paramètre	Modifiable	Valeur	Valeur par défaut	Description
Utilisé	Non	True/False	False	Ce paramètre indique si l'objet Drive est utilisé dans le programme.
Adresse	Non	%DRVn	%DRVn	Ce paramètre indique l'adresse de l'objet Drive, où <i>n</i> est le numéro de l'objet.
Symbole	Oui	–	–	Ce paramètre permet de spécifier un symbole à associer à l'objet Drive. Cliquez deux fois sur la cellule pour définir ou modifier un symbole.
Commentaire	Oui	–	–	Ce paramètre permet de spécifier un commentaire à associer à l'objet Drive. Cliquez deux fois sur la cellule pour définir ou modifier un commentaire.

MC_Power_ATV : Activer/désactiver l'étage de puissance

Description

Ce bloc fonction active ou désactive la phase d'alimentation du variateur.

La phase d'alimentation est activée par un front montant sur l'entrée *Enable*. Une fois la phase d'alimentation activée, la sortie *Status* est définie sur 1.

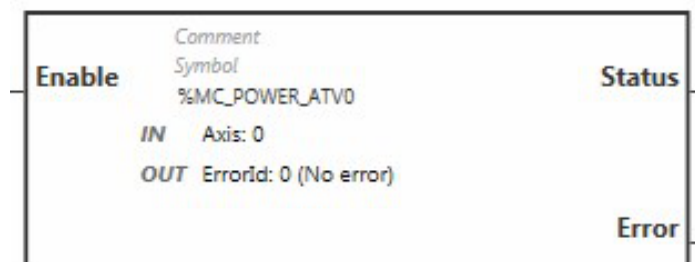
La phase d'alimentation est désactivée par un front descendant sur l'entrée *Enable* (commande *Shutdown* sans *Error*). Une fois la phase d'alimentation désactivée, la sortie *Status* est remise à 0.

Si le registre d'état interne ETA du variateur ATV n'a pas atteint un état de fonctionnement avant expiration de la valeur de timeout, une erreur *Timeout Error* est générée. Le timeout est soit la durée de cycle de la voie multipliée par 4, soit la valeur minimale de 10 secondes. La valeur minimale requise de 10 secondes vise à laisser au variateur le temps de réagir.

Si des erreurs sont détectées au cours de l'exécution du bloc fonction, la sortie *Error* est définie sur 1. Une commande Shutdown (CMD = 16#0006) est alors nécessaire pour désactiver le variateur ATV (état Ready to switch on, ETA = 16#xx21).

Si une erreur se produit, la phase d'alimentation est rétablie uniquement si l'exécution du bloc fonction MC_Reset_ATV, page 72 se déroule correctement.

Représentation graphique



Entrées

Ce tableau décrit les entrées du bloc fonction :

Etiquette	Objet	Valeur initiale	Description
<i>Activer</i>	-	0	Définie sur 1 pour lancer l'exécution du bloc fonction et activer la phase d'alimentation. Définie sur 0 pour arrêter l'exécution du bloc fonction et désactiver la phase d'alimentation.
<i>Axe</i>	%MC_POWER_ATVi.AXIS où i est compris entre 0 et 15	-	Identifiant de l'axe (%DRV0...%DRV15) pour lequel le bloc fonction doit être exécuté.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Etiquette	Objet	Valeur initiale	Valeur
<i>Etat</i>	%MC_POWER_ATVi.STATUS où i est compris entre 0 et 15	0	Valeur par défaut : 0 <ul style="list-style-type: none"> 0 : L'étage de puissance est désactivé. 1 : L'étage de puissance est activé. Défini sur 1 lorsque le variateur ATV atteint un état de fonctionnement (ETA = 16#xx37).
<i>Erreur</i>	%MC_POWER_ATVi.ERROR où i est compris entre 0 et 15	0	Défini sur 0 si aucune erreur n'est détectée. Défini sur 1 si une erreur est détectée en cours d'exécution. L'exécution du bloc fonction est terminée. L'objet de sortie <i>ErrorId</i> indique la cause de l'erreur.
<i>ErrorId</i>	%MC_POWER_ATVi.ERRORID où i est compris entre 0 et 15	0 (aucune erreur)	Code d'erreur renvoyé par le bloc fonction lorsque la sortie <i>Error</i> est définie sur 1. Pour plus d'informations sur les erreurs, consultez les Codes d'erreur, page 75. Plage : 0 à 65535

Paramètres

Cliquez deux fois sur le bloc fonction pour afficher ses paramètres.

Le bloc fonction *MC_Power_ATV* a les paramètres suivants :

Paramètre	Valeur	Description
Utilisation	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	%MC_Power_ATVi	Identificateur de l'instance, i étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets Drive, consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert – Basic – Guide d'exploitation).
Axe	%DRVn, n étant compris entre 0 et 15 Aucun	Sélectionnez l'axe (instance d'objet Drive) pour lequel le bloc fonction doit être exécuté. L'objet Drive doit avoir été configuré au préalable sur Modbus TCP IOScanner ou le scrutateur d'E/S Modbus Série (voir Modicon M221 Logic Controller - Guide de programmation).
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Cliquez deux fois dans la colonne Commentaire et tapez le commentaire de votre choix.

Actualisez les paramètres et cliquez sur **Appliquer**.

MC_Jog_ATV : Démarrer le mode Jog

Description

Ce bloc fonction démarre le mode de fonctionnement Jog. Une opération Jog commande un équipement pour qu'il effectue un mouvement dans le sens positif ou négatif à une vitesse spécifiée.

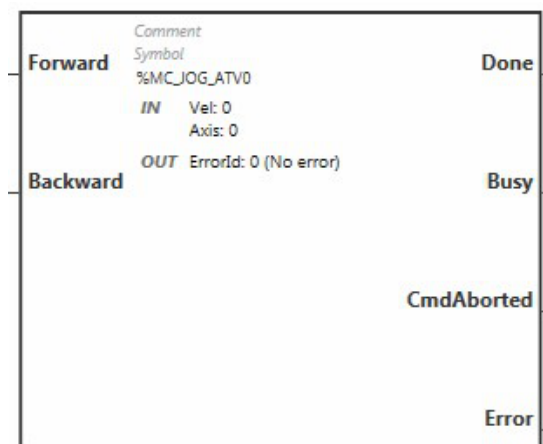
Si un des blocs fonction *MC_MoveVel_ATV*, page 63 ou *MC_Stop_ATV*, page 66 est activé au cours de l'exécution de ce bloc fonction (sortie *Busy* définie sur 1), le bloc fonction *MC_Jog_ATV* commande le mouvement. La sortie *Busy* est remise à 0 et la sortie *CmdAborted* passe à 1.

Lorsqu'une opération Jog est en cours, une modification de la valeur de vitesse (*Vel*) n'est appliquée qu'à la détection d'un front montant ou descendant des entrées *Forward* ou *Backward*.

Si l'une des sorties *Error* ou *CmdAborted* est définie sur 1, les entrées *Forward* et *Backward* doivent d'abord être remises à 0, puis un nouveau front montant doit être appliqué aux entrées *Forward* et/ou *Backward* pour que le mouvement soit redémarré.

Le démarrage d'une opération Jog pendant l'exécution du bloc fonction *MC_Stop_ATV*, page 66 produit une erreur Stop Active Error. Le démarrage d'une opération Jog alors que le variateur n'est pas dans un état de fonctionnement (ETA ≠ 16#xx37) produit une erreur Not Run Error.

Représentation graphique



Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Objet	Valeur initiale	Description
<i>Forward</i>	-	0	Si l'entrée <i>Forward</i> ou <i>Backward</i> est définie sur 1, le mouvement par incréments (jog) démarre. Si les entrées <i>Forward</i> et <i>Backward</i> sont à l'état 1, le mode de fonctionnement reste actif, le mouvement par incréments (jog) est arrêté et la sortie <i>Busy</i> reste définie sur 1. Si les deux entrées <i>Forward</i> et <i>Backward</i> sont définies sur 0, le mode de fonctionnement est arrêté et la sortie <i>Done</i> est définie sur 1 pour un cycle.
<i>Backward</i>	-	0	
<i>Vel</i>	%MC_JOG_ATVi.VEL où i est compris entre 0 et 15	0	Vitesse cible pour le mode de fonctionnement Jog, en tours par minute (tr/min). Lorsqu'un mouvement Jog est en cours, une modification de la valeur de vitesse <i>Vel</i> n'est appliquée qu'à la détection d'un front montant ou descendant des entrées <i>Forward</i> ou <i>Backward</i> . Plage : -32768 à 32767
<i>Axis</i>	%MC_JOG_ATVi.AXIS où i est compris entre 0 et 15	-	Identifiant de l'axe (%DRV0...%DRV15) pour lequel le bloc fonction doit être exécuté. L'axe doit être déclaré au préalable dans l'onglet Configuration .

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Objet de sortie	Valeur initiale	Description
<i>Done</i>	<i>%MC_JOG_ATVi.DONE</i>	0	Définie sur 1 pour un cycle lorsque les deux entrées <i>Forward</i> et <i>Backward</i> sont définies sur 0. Définie sur 1 pour indiquer que le mode de fonctionnement Jog est arrêté.
<i>Busy</i>	<i>%MC_JOG_ATVi.BUSY</i>	0	Réglée sur 1 dans les cas suivants : <ul style="list-style-type: none"> • <i>Jog</i> est en cours (<i>Forward</i> = 1 ou <i>Backward</i> = 1) • les deux entrées <i>Forward</i> et <i>Backward</i> sont définies sur 1, indiquant ainsi que le mode de fonctionnement Jog demeure actif et que le mouvement par incréments est arrêté.
<i>CmdAborted</i>	<i>%MC_JOG_ATVi.CMDABORTED</i>	0	Définie sur 1 si l'exécution d'une autre commande arrête l'exécution du bloc fonction.
<i>Error</i>	<i>%MC_JOG_ATVi.ERROR</i>	0	Défini sur 0 si aucune erreur n'est détectée. Défini sur 1 si une erreur est détectée en cours d'exécution. L'exécution du bloc fonction est terminée. L'objet de sortie <i>ErrorId</i> indique la cause de l'erreur.
<i>ErrorId</i>	<i>%MC_JOG_ATVi.ERRORID</i>	0 (aucune erreur)	Code d'erreur renvoyé par le bloc fonction lorsque la sortie <i>Error</i> est définie sur 1. Pour plus d'informations sur les erreurs, consultez les Codes d'erreur, page 75. Plage : 0 à 65535

Paramètres

Cliquez deux fois sur le bloc fonction pour afficher ses paramètres.

Le bloc fonction *MC_Jog_ATV* a les paramètres suivants :

Paramètre	Valeur	Description
Utilisation	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	<i>%MC_Jog_ATVi</i>	Identificateur de l'instance, <i>i</i> étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets Drive, consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert – Basic – Guide d'exploitation).
Axe	<i>%DRV_n</i> , <i>n</i> étant compris entre 0 et 15 Aucun	Sélectionnez l'axe (instance d'objet Drive) pour lequel le bloc fonction doit être exécuté. L'objet Drive doit avoir été configuré au préalable sur Modbus TCP IOScanner ou le scrutateur d'E/S Modbus Série (voir Modicon M221 Logic Controller - Guide de programmation).
Vel	Vitesse cible	Entrez la vitesse cible du mode de fonctionnement Jog et appuyez sur Entrée. Valeur par défaut : 0 Plage : -32768 à 32767
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Cliquez deux fois dans la colonne Commentaire et tapez le commentaire de votre choix.

Actualisez les paramètres et cliquez sur **Appliquer**.

MC_MoveVel_ATV : Déplacement à une vitesse spécifiée

Description

Ce bloc fonction démarre le mode de fonctionnement Profile Velocity à une vitesse spécifiée. Une fois la vitesse cible atteinte, la sortie *InVel* est définie sur 1.

Si les blocs fonction *MC_Jog_ATV*, page 60 ou *MC_Stop_ATV*, page 66 sont activés au cours de l'exécution de ce bloc fonction (sortie *Busy* définie sur 1), le bloc fonction *MC_MoveVel_ATV* commande le mouvement. Dans ce cas, la sortie *Busy* est remise à 0 et la sortie *CmdAborted* passe à 1.

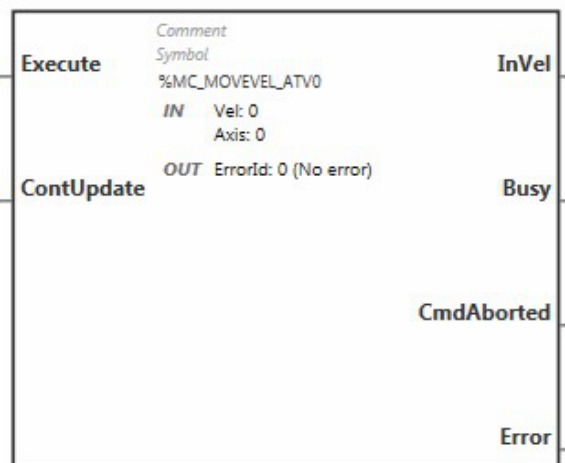
Les valeurs d'entrée *ContUpdate* et *Vel* sont appliquées sur un front montant de l'entrée *Execute*.

Si la sortie *Error* ou *CmdAborted* du bloc fonction *MC_MoveVel_ATV* est définie sur 1, un nouveau front montant est nécessaire sur l'entrée *Execute* pour que le mouvement puisse reprendre.

Le démarrage de ce bloc fonction pendant l'exécution du bloc fonction *MC_Stop_ATV*, page 66 produit une erreur Stop Active Error.

Le démarrage de ce bloc fonction alors que le variateur n'est pas dans un état de fonctionnement (ETA ≠ 16#xx37) produit une erreur Not Run Error.

Représentation graphique



Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Objet	Valeur initiale	Description
<i>Execute</i>	-	0	Définie sur 1 pour lancer l'exécution du bloc fonction.
<i>ContUpdate</i>	-	0	Définie sur 1 avant l'exécution du bloc fonction pour activer la mise à jour en continu de la valeur du paramètre <i>Vel</i> .
<i>Vel</i>	%MC_MOVEVEL_ ATVi.VEL où i est compris entre 0 et 15	0	Vitesse cible pour le mode de fonctionnement, en tours par minute (tr/min). Plage : -32768 à 32767. Une valeur négative force le mouvement dans le sens opposé.
<i>Axis</i>	%MC_MOVEVEL_ ATVi.AXIS où i est compris entre 0 et 15	-	Identifiant de l'axe (%DRV0...%DRV15) pour lequel le bloc fonction doit être exécuté. L'axe doit être déclaré au préalable dans l'onglet Configuration .

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Objet	Valeur initiale	Description
<i>InVel</i>		0	0 indique que la vitesse cible (<i>Vel</i>) n'a pas été atteinte. Définie sur 1 lorsque la vitesse cible (<i>Vel</i>) est atteinte.
<i>Busy</i>	%MC_MOVEVEL_ATVi.BUSY	0	Définie sur 1 lorsque le bloc fonction est exécuté. Reste définie sur 1 même lorsque la vitesse cible est atteinte. Remise à 0 lorsque le bloc fonction est arrêté ou abandonné.
<i>CmdAborted</i>	%MC_MOVEVEL_ATVi.CMDABORTED	0	Définie sur 1 si l'exécution d'une autre commande arrête l'exécution du bloc fonction.
<i>Error</i>	%MC_MOVEVEL_ATVi.ERROR	0	Défini sur 0 si aucune erreur n'est détectée. Défini sur 1 si une erreur est détectée en cours d'exécution. L'exécution du bloc fonction est terminée. L'objet de sortie <i>ErrorId</i> indique la cause de l'erreur.
<i>ErrorId</i>	%MC_MOVEVEL_ATVi.ERRORID	0 (aucune erreur)	Code d'erreur renvoyé par le bloc fonction lorsque la sortie <i>Error</i> est définie sur 1. Pour plus d'informations sur les erreurs, consultez les Codes d'erreur, page 75. Plage : 0 à 65535

NOTE: Lorsque la commande vitesse du variateur ATV est faible (inférieure à 10), il est possible que les paramètres *InVel* et *ConstantVel* soient non valides car la plage de vitesse du variateur ATV est sans doute inexacte.

Paramètres

Cliquez deux fois sur le bloc fonction pour afficher ses paramètres.

Le bloc fonction *MC_MoveVel_ATV* a les paramètres suivants :

Paramètre	Valeur	Description
Utilisation	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	%MC_MoveVel_ATVi	Identificateur de l'instance, <i>i</i> étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets Drive, consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert – Basic – Guide d'exploitation).
Axe	%DRV n , n étant compris entre 0 et 15 Aucun	Sélectionnez l'axe (instance d'objet Drive) pour lequel le bloc fonction doit être exécuté. L'objet Drive doit avoir été configuré au préalable sur Modbus TCP IOScanner ou le scrutateur d'E/S Modbus Série (voir Modicon M221 Logic Controller - Guide de programmation).
Vel	Vitesse cible	Entrez la vitesse cible du mode de fonctionnement et appuyez sur Entrée. Valeur par défaut : 0 Plage : -32768 à 32767. Une valeur négative force le mouvement dans le sens opposé.
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Cliquez deux fois dans la colonne Commentaire et tapez le commentaire de votre choix.

Actualisez les paramètres et cliquez sur **Appliquer**.

MC_Stop_ATV : Arrêt de mouvement

Description

Ce bloc fonction arrête le mouvement en cours du variateur spécifié.

Les paramètres d'arrêt spécifiques au variateur, tels que la décélération, sont fournis par la configuration du variateur.

Une fois que le bloc fonction est démarré par un front montant sur l'entrée *Execute*, toute activité ultérieure sur l'entrée *Execute* est ignorée jusqu'à ce que la sortie *Done* soit définie sur TRUE. L'exécution d'un autre bloc fonction Drive pendant que *MC_Stop_ATV* est à l'état Busy n'annule pas la procédure d'arrêt. Le bloc fonction *MC_Stop_ATV* reste à l'état Busy et l'autre bloc fonction donne une erreur.

La procédure d'arrêt peut être interrompue uniquement par la désactivation de la phase d'alimentation ou bien en cas d'erreur (par exemple, une erreur ATV Not Run ou une erreur du Modbus TCP IOScanner ou du scrutateur d'E/S Modbus Série).

Représentation graphique



Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Objet	Valeur initiale	Description
<i>Exécuter</i>	-	0	Définie sur 1 pour lancer l'exécution du bloc fonction. Il est impossible d'exécuter un autre bloc fonction de mouvement lorsque la sortie <i>Busy</i> est définie sur 1. Dans ce cas, l'autre bloc fonction renvoie une erreur.
<i>Axe</i>	<i>%MC_STOP_ATVi.AXIS</i> où i est compris entre 0 et 15	-	Identifiant de l'axe (<i>%DRV0...%DRV15</i>) pour lequel le bloc fonction doit être exécuté.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Objet de sortie	Valeur initiale	Description
<i>Done</i>	<i>%MC_STOP_ATVi.DONE</i>	0	Définie sur 1 pour indiquer que l'exécution du bloc fonction s'est déroulée correctement.
<i>Busy</i>	<i>%MC_STOP_ATVi.BUSY</i>	0	Définie sur 1 lorsque l'exécution du bloc fonction démarre.
<i>Error</i>	<i>%MC_STOP_ATVi.ERROR</i>	0	Défini sur 0 si aucune erreur n'est détectée. Défini sur 1 si une erreur est détectée en cours d'exécution. L'exécution du bloc fonction est terminée. L'objet de sortie <i>ErrorId</i> indique la cause de l'erreur.
<i>ErrorId</i>	<i>%MC_STOP_ATVi.ERRORID</i>	0 (aucune erreur)	Code d'erreur renvoyé par le bloc fonction lorsque la sortie <i>Error</i> est définie sur 1. Pour plus d'informations sur les erreurs, consultez les Codes d'erreur, page 75. Plage : 0 à 65535

Paramètres

Cliquez deux fois sur le bloc fonction pour afficher ses paramètres.

Le bloc fonction *MC_Stop_ATV* a les paramètres suivants :

Paramètre	Valeur	Description
Utilisation	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	<i>%MC_Stop_ATVi</i>	Identificateur de l'instance, <i>i</i> étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets Drive, consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert – Basic – Guide d'exploitation).
Axe	<i>%DRVn</i> , <i>n</i> étant compris entre 0 et 15 Aucun	Sélectionnez l'axe (instance d'objet Drive) pour lequel le bloc fonction doit être exécuté. L'objet Drive doit avoir été configuré au préalable sur Modbus TCP IOScanner ou le scrutateur d'E/S Modbus Série (voir Modicon M221 Logic Controller - Guide de programmation).
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Cliquez deux fois dans la colonne Commentaire et tapez le commentaire de votre choix.

Actualisez les paramètres et cliquez sur **Appliquer**.

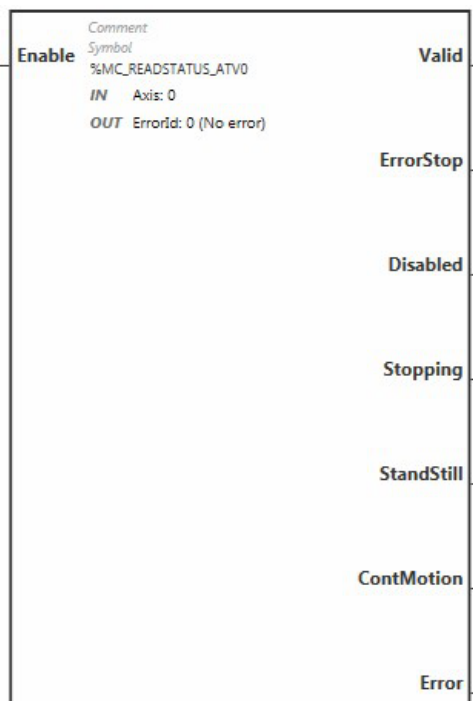
MC_ReadStatus_ATV : Lire l'état de l'équipement

Description

Ce bloc fonction lit l'état du variateur ATV.

Pour plus d'informations sur les états, consultez le Schéma des états du variateur, page 55.

Représentation graphique



Entrées

Ce tableau décrit les entrées du bloc fonction :

Etiquette	Objet	Valeur initiale	Description
Activer	-	0	Définie sur 1 pour activer le bloc fonction.
Axe	%MC_READSTATUS_ATVi. AXIS où i est compris entre 0 et 15	-	Identifiant de l'axe (%DRV0...%DRV15) pour lequel le bloc fonction doit être exécuté.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Etiquette	Objet	Valeur initiale	Description
<i>Valid</i>	%MC_READSTATUS_ATVi. VALID	0	Définie sur 1 lorsque le bloc fonction s'exécute sans erreurs.
<i>ErrorStop</i>	%MC_READSTATUS_ATVi. ERRORSTOP	0	Définie sur 1 si le variateur ATV est dans un état d'erreur (ETA = 16#xxx8).
<i>Disabled</i>	%MC_READSTATUS_ATVi. DISABLED	0	Définie sur 1 si le variateur ATV n'est ni dans un état de fonctionnement ni dans un état d'erreur.
<i>Stopping</i>	%MC_READSTATUS_ATVi. STOPPING	0	Définie sur 1 si le bloc fonction MC_Stop_ATV est en cours d'exécution ou lorsque le mouvement est en cours d'arrêt.
<i>Standstill</i>	%MC_READSTATUS_ATVi. STANDSTILL	0	Définie sur 1 si le variateur ATV est dans un état de fonctionnement et la vitesse est égale à 0 (ETA = 16#xx37 et RFRD = 0).
<i>ContMotion</i>	%MC_READSTATUS_ATVi. CONTMOTION	0	Définie sur 1 si le variateur ATV est dans un état de fonctionnement et la vitesse est différente de 0 (ETA = 16#xx37 et RFRD ≠ 0).
<i>Error</i>	%MC_READSTATUS_ATVi. ERROR	0	Défini sur 0 si aucune erreur n'est détectée. Défini sur 1 si une erreur est détectée en cours d'exécution. L'exécution du bloc fonction est terminée. L'objet de sortie <i>ErrorId</i> indique la cause de l'erreur.
<i>ErrorId</i>	%MC_READSTATUS_ATVi. ERRORID	0 (aucune erreur)	Code d'erreur renvoyé par le bloc fonction lorsque la sortie <i>Error</i> est définie sur 1. Pour plus d'informations sur les erreurs, consultez les Codes d'erreur, page 75. Plage : 0 à 65535

Paramètres

Cliquez deux fois sur le bloc fonction pour afficher ses paramètres.

Le bloc fonction *MC_ReadStatus_ATV* a les paramètres suivants :

Paramètre	Valeur	Description
Utilisation	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	%MC_ReadStatus_ATVi	Identificateur de l'instance, i étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets Drive, consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert – Basic – Guide d'exploitation).
Axe	%DRVn, n étant compris entre 0 et 15 Aucun	Sélectionnez l'axe (instance d'objet Drive) pour lequel le bloc fonction doit être exécuté. L'objet Drive doit avoir été configuré au préalable sur Modbus TCP IOScanner ou le scrutateur d'E/S Modbus Série (voir Modicon M221 Logic Controller - Guide de programmation).
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Cliquez deux fois dans la colonne Commentaire et tapez le commentaire de votre choix.

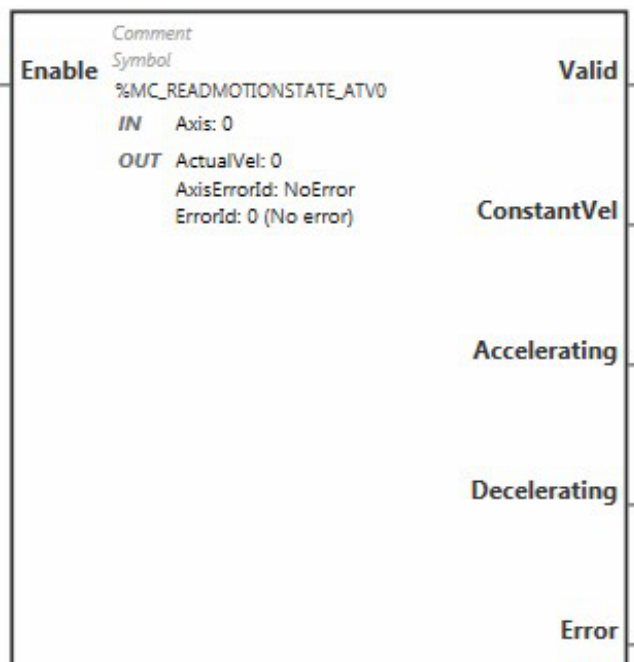
Actualisez les paramètres et cliquez sur **Appliquer**.

MC_ReadMotionState_ATV : Lire l'état de mouvement

Description

Ce bloc fonction produit des informations d'état concernant le mouvement lu à partir du variateur ATV.

Représentation graphique



Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Objet	Valeur initiale	Description
Enable	-	0	Définie sur 1 pour lancer l'exécution du bloc fonction.
Axis	%MC_READMOTIONSTATE_ATVi. AXIS où i est compris entre 0 et 15	-	Identifiant de l'axe (%DRV0...%DRV15) pour lequel le bloc fonction doit être exécuté.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Objet	Valeur initiale	Description
<i>Valid</i>	%MC_READMOTIONSTATE_ATVi.VALID	0	Définie sur 1 lorsque le bloc fonction s'exécute sans erreurs.
<i>ConstantVel</i>	%MC_READMOTIONSTATE_ATVi.CONSTANTVEL	0	Définie sur 1 lorsqu'un mouvement à vitesse constante est effectué (registre ETA).
<i>Accelerating</i>	%MC_READMOTIONSTATE_ATVi.ACCELERATING	0	Définie sur 1 en cas d'accélération du moteur (registre ETI).
<i>Decelerating</i>	%MC_READMOTIONSTATE_ATVi.DECCELERATING	0	Définie sur 1 en cas de décélération du moteur (registre ETI).
<i>Error</i>	%MC_READMOTIONSTATE_ATVi.ERROR	0	Défini sur 0 si aucune erreur n'est détectée. Défini sur 1 si une erreur est détectée en cours d'exécution. L'exécution du bloc fonction est terminée. L'objet de sortie <i>ErrorId</i> indique la cause de l'erreur.
<i>ActualVel</i>	%MC_READMOTIONSTATE_ATVi.ACTUALVEL	0	Il s'agit de la vitesse renvoyée par le variateur ATV (registre RFRD). Plage : -32768 à 32767
<i>AxisErrorId</i>	%MC_READMOTIONSTATE_ATVi.AXISERRORID	0	Il s'agit de l'identifiant d'erreur d'axe renvoyé par le variateur ATV (registre DP0). Une erreur d'axe se produit lorsque le variateur est dans un état d'erreur. Définie sur 0 si le variateur n'est pas dans un état d'erreur (registre ETA ≠ 16#xxx8). Pour plus d'informations sur les erreurs d'axe, consultez les Codes d'erreur AxisErrorId, page 75. Plage : -32768 à 32767
<i>ErrorId</i>	%MC_READMOTIONSTATE_ATVi.ERRORID	Aucune erreur (nOF)	Code d'erreur renvoyé par le bloc fonction lorsque la sortie <i>Error</i> est définie sur 1. Pour plus d'informations sur les erreurs, consultez les Codes d'erreur, page 75. Plage : 0 à 65535

NOTE: Lorsque la commande vitesse du variateur ATV est faible (inférieure à 10), il est possible que les paramètres *InVel* et *ConstantVel* soient non valides car la plage de vitesse du variateur ATV est sans doute inexacte.

Paramètres

Cliquez deux fois sur le bloc fonction pour afficher ses paramètres.

Le bloc fonction *MC_ReadMotionState_ATV* a les paramètres suivants :

Paramètre	Valeur	Description
Utilisation	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	%MC_ReadMotionState_ATVi	Identificateur de l'instance, i étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets Drive, consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert – Basic – Guide d'exploitation).
Axe	%DRVn, n étant compris entre 0 et 15 Aucun	Sélectionnez l'axe (instance d'objet Drive) pour lequel le bloc fonction doit être exécuté. L'objet Drive doit avoir été configuré au préalable sur Modbus TCP IOScanner ou le scrutateur d'E/S Modbus Série (voir Modicon M221 Logic Controller - Guide de programmation).
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Cliquez deux fois dans la colonne Commentaire et tapez le commentaire de votre choix.

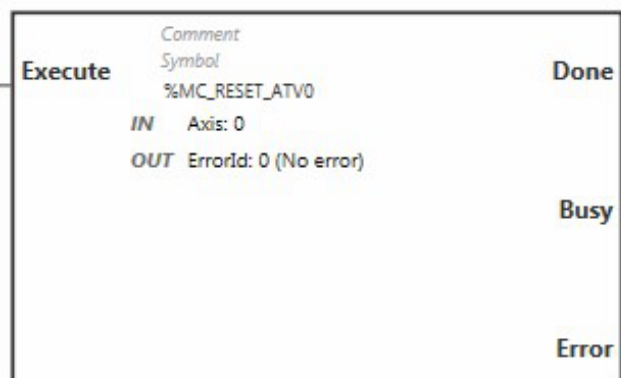
Actualisez les paramètres et cliquez sur **Appliquer**.

MC_Reset_ATV : Acquiescement et réinitialisation d'erreur

Description

Ce bloc fonction permet d'acquiescer une erreur et de réinitialiser la condition d'erreur sur le variateur. Pour plus d'informations, consultez le Schéma des états du variateur, page 55.

Représentation graphique



Entrées

Ce tableau décrit les entrées du bloc fonction :

Etiquette	Objet	Valeur initiale	Description
<i>Exécuter</i>	-	0	Définie sur 1 pour lancer l'exécution du bloc fonction.
<i>Axe</i>	<i>%MC_RESET_ATVi.AXIS</i> où i est compris entre 0 et 15	-	Identifiant de l'axe (<i>%DRV0...%DRV15</i>) pour lequel le bloc fonction doit être exécuté.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Objet de sortie	Valeur initiale	Description
<i>Terminé</i>	<i>%MC_RESET_ATVi.DONE</i>	0	Définie sur 1 lorsque <i>Reset</i> se termine sans erreurs.
<i>Occupé</i>	<i>%MC_RESET_ATVi.BUSY</i>	0	Définie sur 1 lorsque l'exécution du bloc fonction démarre.
<i>Erreur</i>	<i>%MC_RESET_ATVi.ERROR</i>	0	Définie sur 1 si l'équipement reste dans un état d'erreur après expiration de la valeur de timeout. La valeur de timeout est la durée de cycle de la voie multipliée par 4 ou 200 ms (retenir la valeur la plus grande). La valeur minimale requise de 200 ms laisse un temps de réaction au variateur. Pour plus d'informations sur la configuration du temps de cycle des voies, reportez-vous à la section Configuration des voies (voir Modicon M221 Logic Controller - Guide de programmation).
<i>ErrorId</i>	<i>%MC_RESET_ATVi.ERRORID</i>	0 (aucune erreur)	Code d'erreur renvoyé par le bloc fonction lorsque la sortie <i>Error</i> est définie sur 1. Pour plus d'informations sur les erreurs, consultez les Codes d'erreur, page 75. Plage : 0 à 65535

Paramètres

Cliquez deux fois sur le bloc fonction pour afficher ses paramètres.

Le bloc fonction *MC_Reset_ATV* a les paramètres suivants :

Paramètre	Valeur	Description
Utilisation	Adresse utilisée	Si ce paramètre est sélectionné, cette adresse est utilisée dans un programme.
Adresse	<i>%MC_Reset_ATVi</i>	Identificateur de l'instance, i étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets Drive, consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert – Basic – Guide d'exploitation).
Axe	<i>%DRVn</i> , n étant compris entre 0 et 15 Aucun	Sélectionnez l'axe (instance d'objet Drive) pour lequel le bloc fonction doit être exécuté. L'objet Drive doit avoir été configuré au préalable sur Modbus TCP IOScanner ou le scrutateur d'E/S Modbus Série (voir Modicon M221 Logic Controller - Guide de programmation).
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Cliquez deux fois dans la colonne Commentaire et tapez le commentaire de votre choix.

Actualisez les paramètres et cliquez sur **Appliquer**.

Codes d'erreur

Codes d'erreur ErrorId

Le tableau suivant indique les codes d'erreur possibles des blocs fonction :

Valeur	Nom	Description
0	Aucune erreur	Aucune erreur détectée.
1	Erreur du scrutateur d'E/S	Erreur détectée sur le scrutateur d'E/S ⁽¹⁾ .
2	ATV est dans un état d'erreur	Le variateur ATV est dans un état d'erreur (ETA = 16#xxx8).
3	Erreur de timeout	La valeur de timeout a expiré avant que le bloc fonction <i>MC_Power_ATV</i> reçoive l'état correct du variateur.
4	Etat ATV non valide	La valeur ETA du variateur ATV est non valide.
5	Erreur de réinitialisation	Le bloc fonction <i>MC_Reset_ATV</i> est requis alors que le variateur ATV est dans un état d'erreur.
6	Erreur d'arrêt du processus actif	Le bloc fonction <i>MC_Jog_ATV</i> ou <i>MV_MoveVelocity_ATV</i> est requis alors que <i>MC_Stop</i> est actif.
7	Erreur d'ATV pas en mode Run	Le bloc fonction <i>MC_Jog_ATV</i> ou <i>MV_MoveVelocity_ATV</i> est requis alors que le variateur ATV n'est pas en état de fonctionnement.
8	Erreur de référence d'axe non valide	L'entrée <i>AxisRef %DRV</i> du bloc fonction n'est pas valide (absente de la configuration du scrutateur d'E/S Modbus TCP IOScanner ou du scrutateur d'E/S Modbus Série (voir Modicon M221 Logic Controller - Guide de programmation)).
9	Erreur interne.	Une erreur de micrologiciel s'est produite.

(1) Uniquement pour Modbus TCP IOScanner.

Si le bloc fonction *%MC_Power_ATV* génère une erreur du scrutateur d'E/S lors de la scrutation de l'équipement, cela peut être dû à une surcharge du réseau Ethernet. Pour identifier la cause de l'erreur, vous pouvez :

- Vérifier l'état du scrutateur d'E/S : %SW212
- Vérifier l'état du variateur : %IWNS (300+x)
- Vérifier l'état de la voie : %IWNS (300+x).y.
- Augmenter le **timeout de réponse** du variateur.

Codes d'erreur AxisErrorId

Le tableau suivant indique les codes d'erreur d'axe de bloc fonction possibles renvoyés par le bloc fonction *MC_ReadMotionStatus* :

Valeur	Nom
0	Aucune erreur (nOF)
2	EEPROM contrôle (EEF1)
3	Configuration incorrecte (CFF)
4	Configuration non valide (CFI)
5	Interruption de communication Modbus (SLF1)
6	Erreur de liaison interne (ILF)
7	Interruption de communication bus de terrain (CnF)
8	Erreur externe (EPF1)
9	Surintensité (OCF)

Valeur	Nom
10	Condensateur de précharge (CrF)
13	Perte 4-20 mA AI2 (LFF2)
15	Surchauffe en entrée (IHF)
16	Surchauffe variateur (OHF)
17	Surcharge moteur (OLF)
18	Surtension bus DC (ObF)
19	Surtension de l'alimentation (OSF)
20	Perte phase sortie unique (OPF1)
21	Perte phase entrée (PHF)
22	Sous-tension de l'alimentation (USF)
23	Court-circuit moteur (SCF1)
24	Survitesse moteur (SOF)
25	Erreur d'auto-tuning
26	Erreur interne 1 (InF1)
27	Erreur interne 2 (InF2)
28	Erreur interne 3 (InF3)
29	Erreur interne 4 (InF4)
30	EEPROM puissance mémoire ROM (EEF2)
32	Court-circuit terre (SCF3)
33	Perte phase sortie (OPF2)
37	Erreur interne 7 (InF7)
38	Erreur bus de terrain (EPF2)
40	Erreur interne 8 (InF8)
42	Interruption de communication PC (SLF2)
45	Interruption de communication IHM (SLF3)
51	Erreur interne 9 (InF9)
52	Erreur interne 10 (InFA)
53	Erreur interne 11 (InFb)
54	Surchauffe IGBT (tJF)
55	Court-circuit IGBT (SCF4)
56	Court-circuit moteur (SCF5)
60	Erreur interne 12 (InFC)
64	Contacteur en entrée (LCF)
68	Erreur interne 6 (InF6)
69	Erreur interne 14 (InFE)
71	Perte 4-20mA AI3 (LFF3)
72	Perte 4-20mA AI4 (LFF4)
73	Compatibilité cartes (HCF)
77	Erreur transfert Conf (CFI2)
79	Perte 4-20mA AI5 (LFF5)
99	Erreur commutation voie (CSF)
100	Sous-charge processus (ULF)

Valeur	Nom
101	Surcharge processus (OLC)
105	Erreur d'angle (ASF)
106	Perte 4-20mA AI1 (LFF1)
107	Erreur fonction de sécurité (SAFF)
110	Erreur détectée Therm AI2 (tH2F)
111	Erreur capteur thermique AI2 (t2CF)
112	Erreur détectée Therm AI3 (tH3F)
113	Erreur capteur thermique AI3 (t3CF)
114	Erreur démarrage cycle de pompage (PCPF)
119	Erreur faible débit pompe (PLFF)
120	Erreur détectée Therm AI4 (tH4F)
121	Erreur capteur thermique AI4 (t4CF)
122	Erreur détectée Therm AI5 (tH5F)
123	Erreur capteur thermique AI5 (t5CF)
126	Erreur pendant essai (drYF)
127	Erreur rétroaction PID (PFMF)
128	Erreur chargement du programme (PGLF)
129	Erreur exécution du programme (PGrF)
130	Erreur pompe principale (MPLF)
131	Erreur contacteur niveau bas (LCLF)
132	Erreur contacteur niveau haut (LCHF)
142	Erreur interne 16 (InFG)
143	Erreur interne 17 (InFH)
144	Erreur interne 0 (InF0)
146	Erreur interne 13 (InFd)
149	Erreur interne 21 (InFL)
151	Erreur interne 15 (InFF)
152	Erreur mise à jour du micrologiciel (FEr)
153	Erreur interne 22 (InFM)
154	Erreur interne 25 (InFP)
155	Erreur interne 20 (InFF)
157	Erreur interne 27 (InFr)

Bloc fonction Pulse Train Output (%PTO)

Contenu de ce chapitre

Description	78
Configuration	88
Programmation	95
Modes de référencement	97
Paramètres des données	103
Modes de fonctionnement	108
Blocs fonction de mouvement	111
Blocs fonction d' administration	134

Utilisation des blocs fonction Pulse Train Output

Ce chapitre fournit des descriptions et des instructions de programmation concernant l'utilisation des blocs fonction `Pulse Train Output`.

Description

Présentation

Cette section décrit la fonction *Pulse Train Output*.

PTO (Pulse Train Output, sortie à train d'impulsions)

Introduction

La fonction *PTO* de M221 fournit des voies de sortie à train d'impulsions pour un nombre d'impulsions et une vitesse (fréquence) définis. La fonction *PTO* permet de contrôler le positionnement ou la vitesse de variateurs ou moteurs pas à pas linéaires monoaxes indépendants, en mode boucle ouverte. La fonction *PTO* ne reçoit aucun retour de position de la part du processus. Par conséquent, les informations de position doivent être intégrées dans le variateur. Les fonctions *PLS* (impulsion), *PWM* (modulation de largeur d'impulsion), *PTO* (sortie à train d'impulsions) et *FREQGEN* (générateur de fréquence) utilisent les mêmes sorties dédiées. Une seule de ces quatre fonctions peut être utilisée sur la même voie.

Une voie *PTO* peut utiliser des signaux d'interface facultatifs pour le référencement (*Ref*), l'événement (*Probe*), les limites (*LimP*, *LimN*) ou l'interface du variateur (*DriveReady*, *DriveEnable*).

Le décalage de l'origine et la compensation du jeu sont également gérés automatiquement pour améliorer la précision du positionnement. Des diagnostics permettent de surveiller les états.

Fonctions prises en charge

Les voies *PTO* prennent en charge les fonctions suivantes :

- Deux modes de sortie (deux voies pour l'impulsion et la direction, ou une voie pour le mode CW/CCW).
- Mouvements monoaxes (vitesse et position).
- Positionnements relatif et absolu, avec gestion automatique de la direction.
- Accélération et décélération trapézoïdales et courbée en S automatiques.
- Référencement (quatre modes avec compensation de décalage).
- Modification dynamique de l'accélération, de la décélération, de la vitesse et de la position.
- Basculement du mode vitesse au mode position.
- Mise en file d'attente des mouvements (mémoire tampon d'un mouvement).
- Capture de position et déclenchement de mouvement par un événement (à l'aide de l'entrée Probe).
- Compensation de jeu.
- Limites (matérielles et logicielles).
- Diagnostics

NOTE: Les blocs fonction de mouvement , page 111 et les blocs fonction d'administration, page 134 permettent de programmer ces fonctions.

Caractéristiques de la fonction PTO

Une voie *PTO* peut avoir jusqu'à cinq entrées physiques :

- Deux sont affectées à la fonction *PTO* pendant la configuration et sont prises en compte lors d'un front montant sur l'entrée :
 - Entrée Ref
 - Entrée Probe
- Trois sont affectées au bloc fonction *MC_Power_PTO*, page 114. Elles n'ont pas d'affectation fixe (non configurées dans l'écran de configuration) et sont lues avec toutes les autres entrées :
 - Entrée *DriveReady*
 - Entrée Limite positive
 - Entrée Limite négative

NOTE: Ces entrées sont gérées comme n'importe quelle autre entrée normale, mais elles sont utilisées par la fonction *PTO* lorsqu'elles sont affectées au bloc fonction *MC_Power_PTO*, page 114.

NOTE: Les entrées de limite positive et de limite négative sont requises pour éviter la surcourse.

▲ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

- Assurez-vous que des détecteurs de limite matérielle du contrôleur sont intégrés dans la conception et la logique de l'application.
- Montez les détecteurs de limite matérielle du contrôleur de telle sorte que la distance de freinage soit adéquate.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Une voie *PTO* peut avoir jusqu'à trois sorties physiques :

- Deux sorties sont obligatoires pour gérer le mode de sortie de la fonction *PTO*. Elles ont une affectation fixe et doivent être activées par configuration :
 - *CW / CCW* (sens horaire / sens contraire)
 - *Impulsion / Direction*
- L'autre sortie, *DriveEnable*, est associée au bloc fonction *MC_Power_PTO*, page 114. Elle n'a aucune affectation fixe et est écrite à la fin du cycle *MAST* comme les sorties normales.

Les caractéristiques de la fonction *PTO* sont présentées ci-après :

Caractéristique	Valeur
Nombre de voies	2 ou 4 selon le module
Nombre d'axes	1 par voie
Plage de positions	-2 147 483 648 à 2 147 483 647 (32 bits)
Vitesse minimum	0 Hz
Vitesse maximum	100 kHz (pour un cycle de service 40/60 et max. 200 mA)
Pas minimum	1 Hz
Précision sur la vitesse	1 %
Accélération/décélération (min.)	1 Hz/ms
Accélération/décélération (max.)	100 kHz/ms
Décalage de l'origine	-2 147 483 648 à 2 147 483 647 (32 bits)
Plages des limites logicielles	-2 147 483 648 à 2 147 483 647 (32 bits)

Modes de sortie d'impulsion

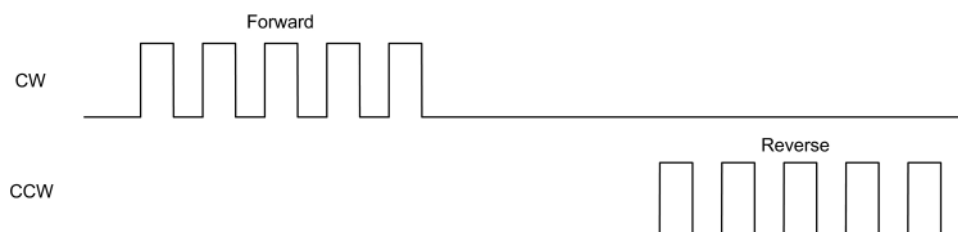
Présentation

Il existe deux modes de sortie possibles :

- Sens des aiguilles d'une montre / Sens contraire
- Impulsion / Direction

Mode Sens horaire (CW) / Sens contraire (CCW)

Ce mode génère un signal qui définit la vitesse de fonctionnement et la direction du moteur. Ce signal est mis en œuvre sur la première voie *PTO* (*PTO0* uniquement).



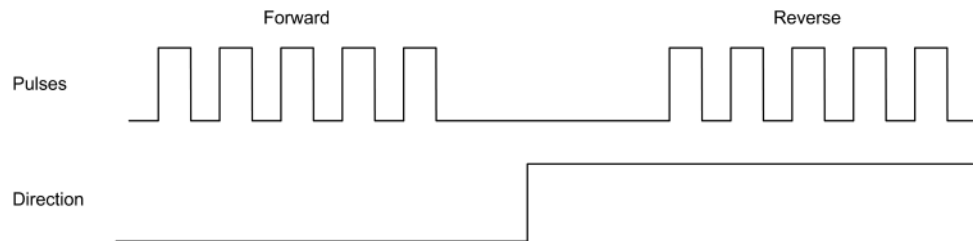
NOTE: *PTO1* n'est pas disponible dans ce mode.

Mode Impulsion / Direction

Ce mode génère deux signaux sur les voies PTO :

- La sortie des impulsions indique la vitesse de fonctionnement du moteur (*Pulses*).
- La sortie de la direction indique le sens de rotation du moteur (*Direction*).

NOTE: Il est possible de désactiver la sortie de la direction si elle n'est pas nécessaire à l'application.



Cas particuliers

Cas particulier	Description
Effet d'un redémarrage à froid (%S0=TRUE)	<ul style="list-style-type: none"> • L'axe est défini à l'état Disabled. • Les blocs fonction PTO sont initialisés.
Effet d'un redémarrage à chaud (%S1=TRUE)	<ul style="list-style-type: none"> • L'axe est défini à l'état Disabled. • Les blocs fonction PTO sont initialisés.
Effet à l'arrêt du contrôleur	<ul style="list-style-type: none"> • L'axe est défini à l'état ErrorStop. • Les sorties sont réinitialisées à 0.
Effet d'une modification en ligne	Aucun(e)

Rampes d'accélération et de décélération

Vitesse de démarrage

La **vitesse de démarrage** est la fréquence minimum à laquelle un moteur pas à pas peut produire un mouvement, avec une charge appliquée, sans aucune perte de pas.

Le paramètre **Vitesse de démarrage** est utilisé lorsqu'un mouvement part de la vitesse 0.

La **vitesse de démarrage** doit être comprise entre 0 et *MaxVelocityAppl*.

La valeur 0 signifie que le paramètre **Vitesse de démarrage** n'est pas utilisé. Dans ce cas, le mouvement débute à une vitesse égale au taux d'accélération x 1 ms.

Vitesse d'arrêt

La **Vitesse d'arrêt** est la fréquence maximum à laquelle un moteur pas à pas peut arrêter un mouvement, avec une charge appliquée, sans aucune perte de pas.

La **vitesse d'arrêt** n'est utilisée pour les mouvements qui partent d'une vitesse supérieure à **Vitesse d'arrêt** pour ralentir jusqu'à la vitesse 0.

La **vitesse d'arrêt** doit être comprise entre 0 et *MaxVelocityAppl*.

La valeur 0 signifie que le paramètre **Vitesse d'arrêt** n'est pas utilisé. Dans ce cas, le mouvement s'arrête à une vitesse égale à taux de décélération x 1 ms.

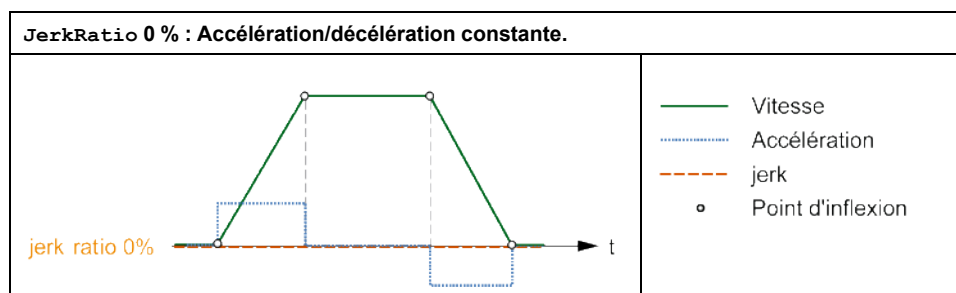
Accélération/Décélération

L'accélération est le taux de variation de la vitesse entre la **Vitesse de départ** et la vitesse cible. La décélération est le taux de variation de la vitesse entre la vitesse cible et la **Vitesse d'arrêt**. Ces changements de vitesse sont gérés de manière implicite par la fonction *PTO* conformément aux paramètres *Acceleration*, *Deceleration* et *JerkRatio* selon un profil **trapézoïdal** ou **courbé en S**.

Rampe d'accélération/décélération avec un profil trapézoïdal

Lorsque le paramètre *JerkRatio* est réglé sur 0, la rampe d'accélération/décélération présente un profil trapézoïdal.

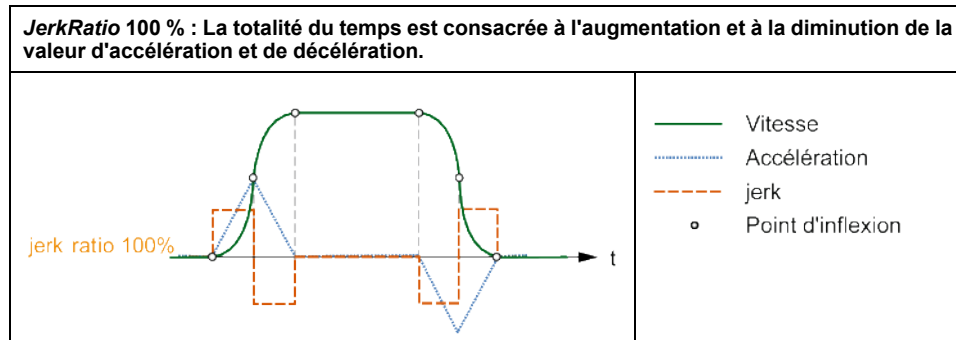
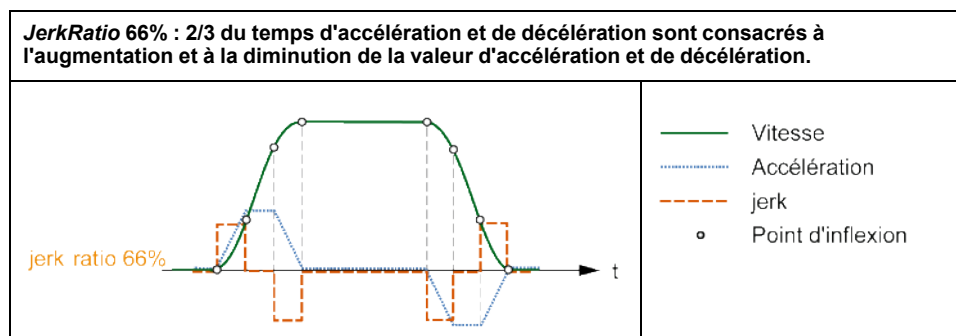
Exprimés en Hz/ms, les paramètres *Acceleration* et *Deceleration* représentent le rythme de changement de la vitesse.



Rampe d'accélération/décélération avec un profil courbé en S

Lorsque le paramètre *JerkRatio* est supérieur à 0, la rampe d'accélération/décélération présente un profil courbé en S.

Le profil courbé en S est utilisé dans les applications contenant une forte inertie ou qui manipulent des liquides ou des objets fragiles. La rampe courbée en S permet une accélération/décélération progressive, comme le montrent les graphiques suivants :



NOTE: La valeur du paramètre *JerkRatio* est commune à l'accélération et à la décélération, de sorte que le temps concave et le temps convexe sont égaux.

Impact de la rampe courbée en S sur l'accélération/décélération

La durée de l'accélération/décélération est maintenue, quel que soit le paramètre *JerkRatio*. Pour conserver cette durée, l'accélération ou la décélération est différente de celle configurée dans le bloc fonction (paramètres *Acceleration* ou *Deceleration*).

Lorsque le *JerkRatio* est appliqué, l'accélération/la décélération est modifiée.

Lorsque le *JerkRatio* est appliqué à 100 %, l'accélération/la décélération est deux fois supérieure à celle des paramètres *Acceleration/Deceleration* configurés.

NOTE: Si la valeur du paramètre *JerkRatio* est non valide, celle-ci est recalculée en fonction des paramètres *MaxAccelerationAppl* et *MaxDecelerationAppl*.

JerkRatio est non valide lorsque :

- Sa valeur est supérieure à 100. Dans ce cas, une valeur de *JerkRatio* égale à 100 est appliquée.
- Sa valeur est inférieure à 0. Dans ce cas, une valeur de *JerkRatio* égale à 0 est appliquée.

Événement de capteur

Description

L'entrée *Probe* est définie par configuration et activée à l'aide du bloc fonction *MC_TouchProbe_PTO*.

L'entrée *Probe* est utilisée comme événement pour :

- capturer la position,
- lancer un mouvement indépendamment de la tâche.

Les deux fonctions peuvent être actives en même temps, c'est-à-dire que le même événement capture la position et démarre un bloc fonction de mouvement, page 78.

NOTE: Seul le premier événement après le front montant sur la sortie *MC_TouchProbe_PTO* du bloc fonction *Busy* est valide. Dès que la sortie *Done* est définie sur TRUE, les événements suivants sont ignorés. Le bloc fonction doit être réactivé pour répondre à d'autres événements.

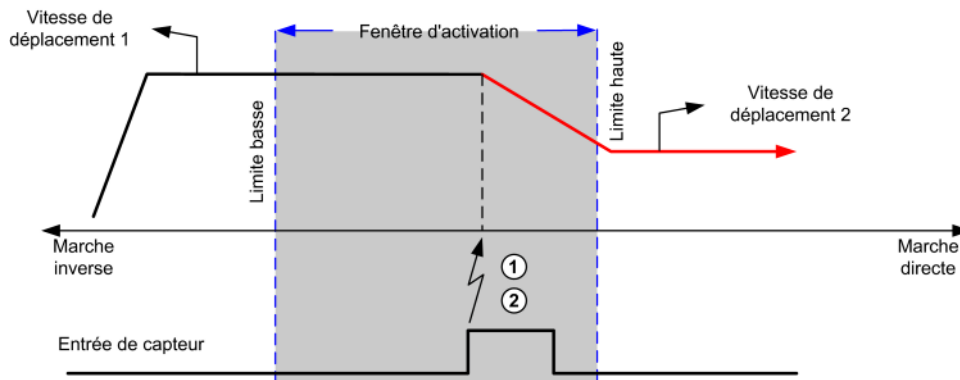
Capture de position

La position capturée est disponible dans *%MC_TouchProbe_PTO.RecordedPos*.

Déclencheur de mouvement

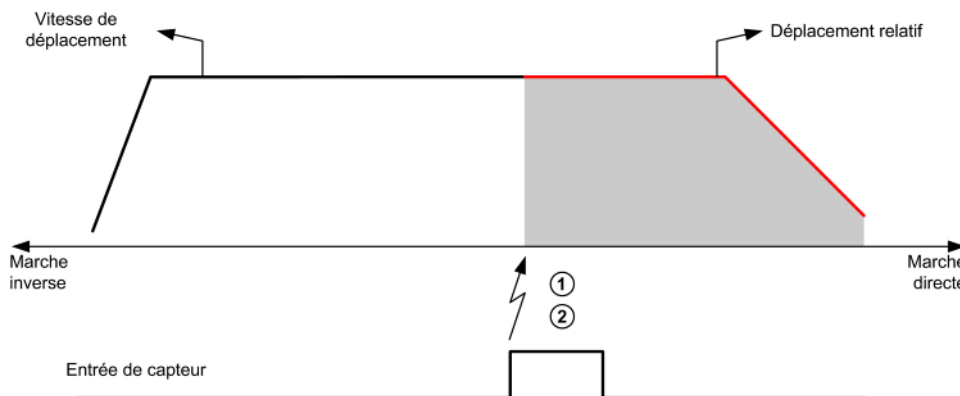
L'entrée *BufferMode* d'un bloc fonction de mouvement doit être définie sur *se Trigger*.

L'exemple suivant illustre un changement de vitesse cible avec fenêtre d'activation :



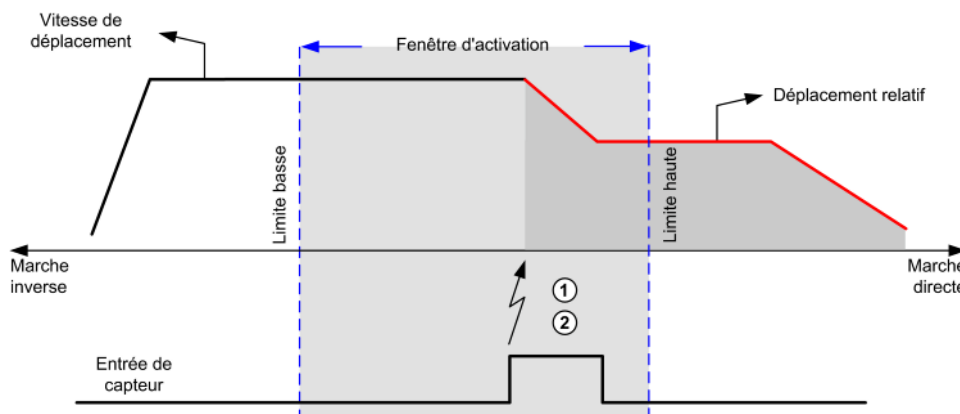
- 1 Capture de la valeur du compteur de position
- 2 Déclenchement du bloc fonction *Move Velocity*

L'exemple suivant illustre un mouvement avec distance pré-programmée, avec un profil simple et pas de fenêtre d'activation :



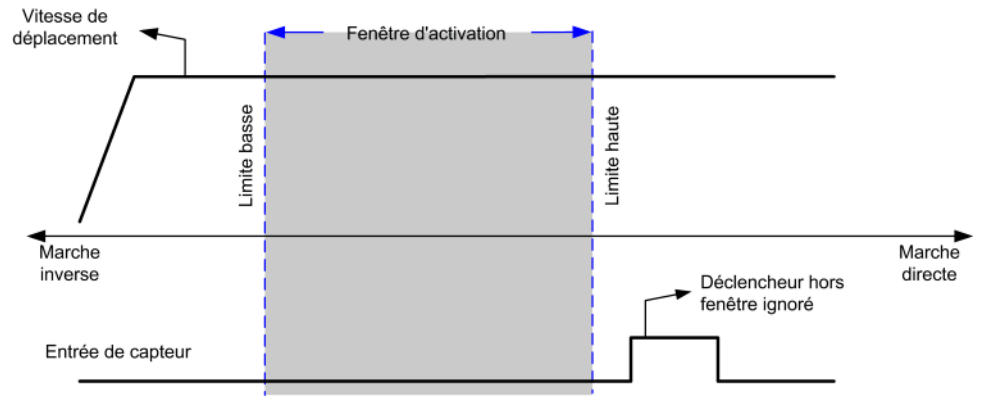
- 1 Capture de la valeur du compteur de position
- 2 Déclenchement du bloc fonction *Move Relative*

L'exemple suivant illustre un mouvement avec distance pré-programmée, avec profil complexe et fenêtre d'activation :



- 1 Capture de la valeur du compteur de position
- 2 Déclenchement du bloc fonction *Move Relative*

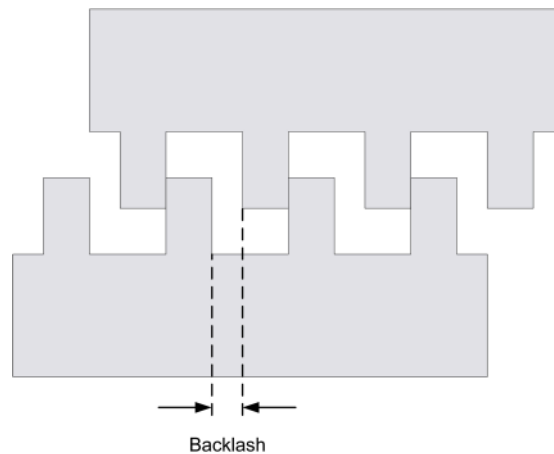
L'exemple suivant illustre un événement déclencheur hors de la fenêtre d'activation :



Compensation de jeu

Description

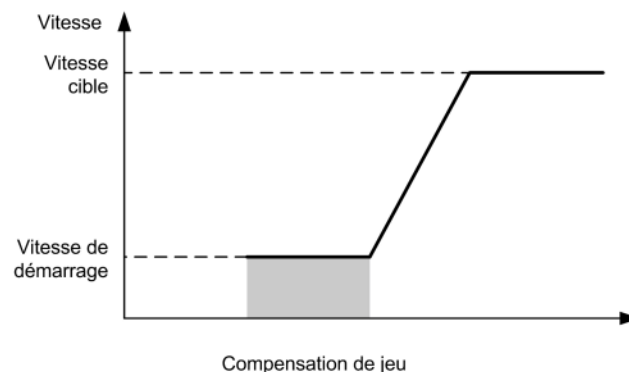
Le paramètre *Backlash Compensation* représente la quantité de mouvement nécessaire pour compenser le jeu mécanique dans les engrenages lorsque la direction est inversée :



NOTE: Cette fonction ne prend pas en compte les sources externes de mouvement, telle que l'inertie ou d'autres formes de mouvement induit.

La compensation du jeu est exprimée en nombre d'impulsions (0 à 65 535, valeur par défaut 0). Lorsqu'elle est définie, à chaque inversion de direction, le nombre spécifié d'impulsions est généré, à la vitesse de démarrage, avant l'exécution du mouvement programmé. Les impulsions de compensation de jeu ne sont pas ajoutées au compteur de position.

La figure suivante illustre la compensation de jeu :



NOTE:

- Avant le démarrage du mouvement initial, la fonction ne peut pas connaître la quantité de jeu à compenser. Par conséquent, la compensation du jeu n'est active qu'après l'exécution d'un premier mouvement et la compensation est appliquée au premier changement de sens.
- Si une commande d'abandon est reçue ou si une erreur est détectée avant la compensation du jeu, la position absolue reste inchangée.
- Après une commande d'abandon, la compensation est redémarrée à partir de la position de jeu en cours lorsqu'un nouveau mouvement est exécuté.

Pour plus d'informations, reportez-vous à la section Configuration de PTO (voir Modicon M221 Logic Controller - Guide de programmation).

Limites de positionnement

Introduction

Il est possible de définir des limites positive et négative pour contrôler l'amplitude du mouvement dans les deux sens. Le contrôleur gère des limites matérielles comme logicielles.

Les détecteurs de limites matérielles et logicielles sont utilisés pour gérer les limites dans l'application contrôleur uniquement. Ils ne sont pas destinés à remplacer les détecteurs de limite de sécurité fonctionnelle raccordés au variateur. Les détecteurs de limite de l'application du contrôleur doivent obligatoirement être activés avant les détecteurs de limite de sécurité fonctionnelle câblés. Dans tous les cas, le type d'architecture de sécurité fonctionnelle que vous pouvez déployer (et qui dépasse le cadre du présent document) dépend de votre analyse de la sécurité, notamment, mais sans limitation :

- évaluation des risques conformément à EN/ISO 12100
- analyse des modes de défaillance et de leurs effets (FMEA) conformément à EN 60812

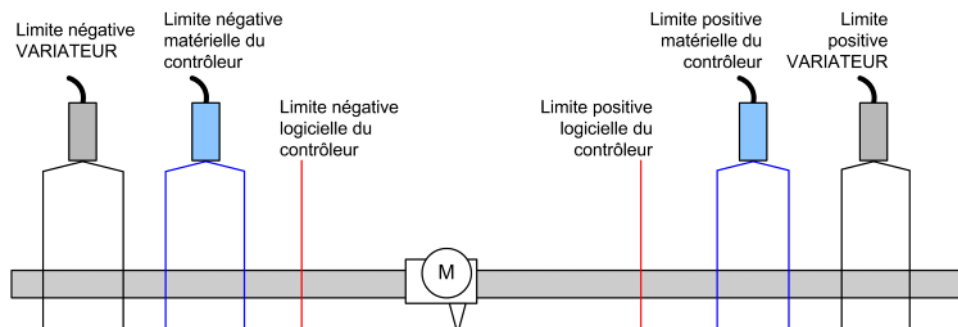
⚠ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

Assurez-vous qu'une évaluation des risques est effectuée et respectée conformément à la norme EN/ISO 12100 pendant la conception de votre machine.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

La figure suivante illustre les détecteurs de limites matérielles et logicielles.

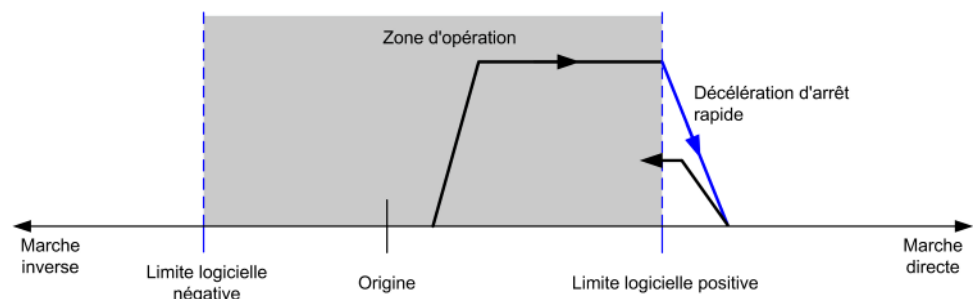


Dès que la limite logicielle négative ou positive est franchie, une erreur est détectée et une décélération d'arrêt rapide se déclenche :

- L'axe passe à l'état *ErrorStop*, avec un code d'erreur *AxisErrorId* de 1002 à 1005. Consultez les sections *MC_ReadAxisError_PTO*, page 140 et Codes d'erreur d'axe PTO, page 106.
- La direction actuelle devient invalide et le paramètre PTO associé *EnableDirPos* (1004) ou *EnableDirNeg* (1005) est remis à 0 par le système.
- le bloc fonction en cours d'exécution détecte l'état d'erreur,
- Sur les autres blocs fonction applicables, les sorties *CmdAborted* sont réglées sur TRUE.

Pour effacer l'état d'erreur de l'axe et revenir à l'état *Standstill*, l'exécution de *MC_Reset_PTO* est requise car toute commande de mouvement est rejetée (reportez-vous aux paramètres PTO, page 105 *EnableDirPos* ou *EnableDirNeg*) tant que l'axe reste en dehors des limites (le bloc fonction se termine avec *ErrorId = InvalidDirectionValue*). Dans ce cas de figure, vous ne pouvez exécuter une commande de mouvement que dans l'autre sens.

Lorsque l'axe est compris dans les limites, le paramètre *EnableDirPos* ou *EnableDirNeg* est restauré sur 1 (valide) par le système.



NOTE: Sur le schéma précédent, le retour de l'axe dans les limites est le résultat de l'exécution de *MC_Reset_PTO* (l'opération n'est pas effectuée automatiquement).

Limites logicielles

Il est possible de définir des limites logicielles pour contrôler l'amplitude du mouvement dans les deux sens.

Les valeurs de ces limites sont activées et définies dans l'écran de configuration et doivent respecter les règles suivantes :

- Limite positive >> limite négative
- Valeurs comprises dans la plage -2 147 483 648 à 2 147 483 647

Elles peuvent également être activées, désactivées ou modifiées dans le programme d'application (*MC_WritePar_PTO* et *PTO_PARAMETER*, page 105).

NOTE: Lorsqu'elles sont activées, les limites logicielles entrent en vigueur après un référencement réussi de l'axe (*MC_Home_PTO*).

Limites matérielles

Les limites matérielles sont nécessaires pour la procédure de référencement et pour éviter d'endommager la machine. Les entrées appropriées doivent être utilisées sur les entrées *%MC_Power_PTO.LimP* et *%MC_Power_PTO.LimN*. Les dispositifs de limite matérielle doivent être de type normalement fermé pour que l'entrée du bloc fonction soit FALSE lorsque la limite correspondante est atteinte.

NOTE: les restrictions de mouvement sont valides lorsque les entrées de limite sont FALSE, quelle que soit la direction. Lorsqu'elles reprennent la valeur TRUE, les restrictions de mouvement sont supprimées et les limites matérielles redeviennent fonctionnelles. Par conséquent, utilisez des contacts de front descendant générant des instructions de sortie RESET avant d'exécuter le bloc fonction. Ensuite, utilisez ces bits pour contrôler les entrées du bloc fonction. Lorsque les opérations sont terminées, configurez les bits pour restaurer un fonctionnement normal.

▲ AVERTISSEMENT

FUNCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

- Assurez-vous que des détecteurs de limite matérielle du contrôleur sont intégrés dans la conception et la logique de l'application.
- Montez les détecteurs de limite matérielle du contrôleur de telle sorte que la distance de freinage soit adéquate.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

NOTE: La distance de freinage appropriée dépend de la vitesse maximum, de la charge (masse) maximum de l'équipement déplacé et de la valeur du paramètre de décélération pour arrêt rapide.

Configuration

Présentation

Cette section explique comment configurer une voie PTO et les paramètres associés.

Configuration de la fonction PTO

Présentation

Pour configurer la ressource *Pulse Generator*, consultez la section Configuration des générateurs d'impulsions (voir Modicon M221, Contrôleur logique, Guide de programmation) dans le document Modicon M221 Logic Controller - Guide de programmation.

Pour configurer la ressource *Pulse Generator* comme un PWM, consultez la section Modicon M221 Logic Controller Configuration du type PWM dans le document (voir Modicon M221, Contrôleur logique, Guide de programmation) - Guide de programmation.

Motion Task Table

Présentation

La Motion Task Table peut être utilisée pour la programmation des blocs fonction de mouvement. Elle est dédiée aux séquences de mouvements répétitifs. Durant la configuration, une séquence de mouvements est définie pour un axe (une séquence peut être comparée à une recette qui associe divers mouvements).

La Motion Task Table peut s'appliquer à plusieurs axes et fournit une représentation graphique de la séquence de mouvements configurée.

Utilisez le bloc fonction *MC_MotionTask_PTO* pour exécuter un Motion Task Table. Lorsque la table est appelée par le bloc fonction *MC_MotionTask_PTO*, elle doit être associée à un axe spécifique. La Motion Task Table est appliquée à l'axe utilisé par le bloc fonction *MC_MotionTask_PTO*. Plusieurs blocs fonction *MC_MotionTask_PTO* peuvent exécuter les mêmes instances %MT Motion Task Table simultanément.

Fonctionnalités

Nombre maximal d'instances Motion Task Table (%MT) : 4

Une Motion Task Table contient une séquence de mouvements monoaxe :

- Une séquence est une succession d'étapes.
- Chaque étape définit les paramètres d'un mouvement.
- Chaque étape utilise une instance de bloc fonction de mouvement dédiée.

Mouvements qui peuvent être utilisés dans la Motion Task Table :

- Mouvement absolu
- Mouvement relatif
- Pause
- Définir position
- Vitesse de mouvement

Configuration d'une Motion Task Table

L'**Assistant Motion Task Table** permet de configurer chaque mouvement dans une séquence ordonnée et de visualiser un profil de mouvement global estimé.

Pour afficher l'**Assistant Motion Task Table**, procédez comme suit :

Etape	Action																														
1	<p>Sélectionnez l'onglet Programmation > Outils et cliquez sur Objets PTO Motion Task Tables dans l'arborescence du matériel pour afficher les propriétés de Motion Task Table.</p> <p>Propriétés tooltipMotionTaskTables</p> <table border="1"> <thead> <tr> <th></th> <th>Configuré</th> <th>Adresse</th> <th>Symbole</th> <th>Configuration</th> <th>Commentaire</th> </tr> </thead> <tbody> <tr> <td></td> <td><input type="checkbox"/></td> <td>%MT0</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td><input type="checkbox"/></td> <td>%MT1</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td><input type="checkbox"/></td> <td>%MT2</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td><input type="checkbox"/></td> <td>%MT3</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		Configuré	Adresse	Symbole	Configuration	Commentaire		<input type="checkbox"/>	%MT0					<input type="checkbox"/>	%MT1					<input type="checkbox"/>	%MT2					<input type="checkbox"/>	%MT3			
	Configuré	Adresse	Symbole	Configuration	Commentaire																										
	<input type="checkbox"/>	%MT0																													
	<input type="checkbox"/>	%MT1																													
	<input type="checkbox"/>	%MT2																													
	<input type="checkbox"/>	%MT3																													
2	Cliquez sur [...] pour configurer le Motion Task Table.																														

Description de la fenêtre des propriétés de Motion Task Table :

Paramètre	Modifiable	Valeur	Valeur par défaut	Description
Configuré	Non	True/False	Faux	Indique si la Motion Task Table contient des étapes configurées.
Adresse	Non	%MTx	%MTx	Indique l'adresse de la Motion Task Table, où x est le numéro de table.
Symbole	Oui	-	-	Permet de définir le symbole à associer à la Motion Task Table. Double-cliquez sur la cellule pour modifier le champ.
Configuration	Oui	[...] (Bouton)	Activé	Permet de configurer la séquence de mouvements en utilisant l' Assistant Motion Task Table .
Commentaire	Oui	-	-	Permet de définir un commentaire à associer à la Motion Task Table. Double-cliquez sur la cellule pour modifier le champ.

Assistant Motion Task Table :

Assistant Motion Task Table
✕

Etapas

Etape	Type	Pos	Distance	Vel	Acc	Dec	Jerk ratio	Etape suivante	Evénement	Delay	Objets logiciels
1	MC_MoveAbs_P	2000		5000	20	50	0	Done		10	%MC_MOVEA
2	MC_MoveRel_P*		5000	7500	20	100	0	Done		0	%MC_MOVER
3	MC_MoveRel_P*		5000	4000	20	200	0	SW event	%M1	1000	%MC_MOVER
4	MC_Halt_PTO					1	0	Done		0	%MC_HAL_PT
5	Aucun										
6	Aucun										

Utiliser une plage d'événement PROBE
 Première position
Dernière position

Présentation du mouvement

? Le graphique ci-dessous peut ne pas représenter des événements réels. Pour plus d'informations, consultez la documentation du produit.

Principales zones de l'**Assistant Motion Task Table** :

Paramètre	Description
Etapes	Séquence des mouvements monoaxes et paramètres d'entrée pour chaque mouvement.
Présentation du mouvement	<p>Cliquez sur le bouton d'actualisation (ou F5) pour générer une vue graphique du mouvement correspondant à la séquence d'étapes.</p> <p>La courbe propose une vue d'ensemble du mouvement. La courbe repose sur les conditions suivantes :</p> <ul style="list-style-type: none">• Position initiale = 0.• Les limites de la position ne sont pas activées.• Les paramètres par défaut de la configuration des mouvements sont appliqués.• Un événement (entrée de capteur, POU) se produit à l'issue de l'étape et dans un délai de 100 ms.• Le délai %MWx représenté sur le graphique est de 100 ms.

Description de la fenêtre **Etapes** :

Paramètre	Valeur	Valeur par défaut	Description
Etape	1...16	–	Numéro du mouvement monoaxe dans la séquence.
Type	Aucun MC_MoveAbs_PTO (Mouvement absolu) MC_MoveRel_PTO (Mouvement relatif) MC_Halt_PTO (Arrêt) MC_SetPos_PTO (Définir la position) MC_MoveVel_PTO (Vitesse de mouvement)	Aucun	Commande de mouvement. La commande de mouvement utilise une instance de bloc de fonction de mouvement indiquée dans le paramètre Objets logiciels .
Pos	Voir la valeur de chaque paramètre du bloc fonction des objets logiciels.	<i>vide</i>	Les paramètres de mouvement sont les paramètres des objets logiciels affectés à l'étape. Description des paramètres : <ul style="list-style-type: none"> • Pos : Position • Distance : Distance • Vel : Vitesse • Acc : Accélération • Dec : Décélération • Jerk ratio : Jerk ratio NOTE: Le paramètre <i>Vel</i> de la commande de vitesse de mouvement est une combinaison de la vitesse et la direction. Dans la table, la plage de vitesse de la commande de mouvement <i>MC_MoveVel_PTO</i> est la suivante : - Vitesse max à + Vitesse Max. Une vitesse négative indique une direction négative, une vitesse positive indique une direction positive.
Distance			
Vel			
Acc			
Dec			
Jerk ratio			
Etape suivante	Done / In velocity Blending previous Probe input event %M event Delay	<i>vide</i>	Condition à remplir pour passer à l'étape suivante dans la séquence de la table. Description de la condition : <ul style="list-style-type: none"> • Done / In velocity : <ul style="list-style-type: none"> ◦ Done : Passer à l'étape suivante une fois l'étape en cours terminée. Ce paramètre est disponible pour toutes les commandes de mouvement, sauf la vitesse de mouvement. ◦ In velocity : Passer à l'étape suivante lorsque la vitesse demandée est atteinte. Ce paramètre est disponible uniquement pour la commande de vitesse de mouvement. • Blending previous : La vitesse de l'étape suivante est fusionnée avec la vitesse de la position finale de cette étape. • Probe input event : Passer à l'étape suivante lorsqu'un événement défini est détecté sur l'entrée Probe. Le front est défini dans le paramètre Evénement. Un champ d'entrée Utiliser une plage d'événement PROBE (voir tableau ci-dessous) s'ouvre en bas de la fenêtre Etapes. NOTE: Une occurrence de Probe input event peut être utilisée par Motion Task Table. • %M event : Passer à l'étape suivante lorsque l'adresse du bit mémoire (<i>%Mx</i>) définie dans le paramètre Evénement est à 1. • Delay : Passer à l'étape suivante lorsque le délai (commençant au début de l'étape) est écoulé. Le délai est défini dans le paramètre Delay. NOTE: Lorsque l'événement Probe input event, %M event ou Delay se produit, l'étape suivante est lancée même si l'étape en cours n'est pas terminée.

Paramètre	Valeur	Valeur par défaut	Description
Événement	– 0/1 %Mx	<i>vide</i>	La valeur de Event complète les conditions décrites dans le paramètre Étape suivante . Sélection de l' Étape suivante et de l' Événement correspondant : <ul style="list-style-type: none"> • Probe input event : <ul style="list-style-type: none"> ◦ 0 : Front descendant ◦ 1 : Front montant • %M event : Bit mémoire %Mx. <p>NOTE : L'événement d'entrée Probe est indépendant du cycle de la tâche d'application et du cycle de la tâche de mouvement.</p> <p>NOTE : %Mx est évalué toutes les 4 ms.</p>
Delay	0...65535 %MWx	<i>vide</i>	La valeur Delay représente le délai avant exécution de l'étape suivante. Selon la valeur du paramètre Étape suivante , la valeur Delay est évaluée à partir du début ou de la fin de l'étape : <ul style="list-style-type: none"> • Done / In velocity : Le délai démarre lorsque l'étape en cours est Done ou In velocity. • Blending previous : Non disponible • Probe input event et %M event : Le délai commence au début de l'étape. <ul style="list-style-type: none"> ◦ A la fin du délai, un timeout est généré si l'événement ne s'est pas produit, et l'étape suivante est effectuée. ◦ Si l'événement se produit avant la fin du délai, l'étape suivante est exécutée et le timeout est annulé. <p>NOTE : Si Delay reste à sa valeur par défaut (0), la commande de mouvement attend que l'entrée de capteur ou l'événement logiciel se produise, sans temporisation.</p> • Delay : Le délai commence au début de l'étape. L'étape suivante est exécutée une fois le délai écoulé. <p>NOTE : Une valeur immédiate ne peut pas être modifiée dans une POU de l'application, mais une valeur %MWx doit être définie par une POU de l'application. Le paramètre Delay de la Motion Task Table n'est pas modifié si <i>MC_ReadPar_PTO</i> ou <i>MC_WritePar_PTO</i> sont définis en utilisant <i>ParNumber</i> = 1000 (delay).</p>
Objets logiciels	%MC_MOVEABS_PTOx %MC_MOVEREL_PTOx %MC_HALT_PTOx %MC_SETPOS_PTOx %MC_MOVEVEL_PTOx	<i>vide</i>	Indique l'objet logiciel alloué à l'étape. Il est alloué par le système et il est en lecture seule. Ces objets logiciels sont des instances de bloc fonction.
Symbole	–	<i>vide</i>	Permet de définir le symbole à associer à l'objet logiciel de l'étape. Double-cliquez sur la cellule pour modifier le champ.

Paramètre **Utiliser une plage d'événement PROBE** dans la fenêtre **Étapes** :

Paramètre	Valeur	Valeur par défaut	Description
Utiliser une plage d'événement PROBE	True/False	Faux	Si la valeur est TRUE, un événement déclencheur n'est reconnu que dans la plage de positions définie entre la Première position et la Dernière position . Le paramètre peut être modifié si l' Étape suivante est définie sur Probe input event dans Motion Task Table.
Dernière position ⁽¹⁾	- 2147483648... 2147483647 %MDx	2147483647	Limite supérieure de la plage de positions.
Première position ⁽¹⁾	- 2147483648... 2147483647 %MDx	- 2147483648	Limite inférieure de la plage de positions.
(1) La valeur de la Première position doit être inférieure à la valeur de la Dernière position . NOTE : L'influence de la plage de positions lors du déclenchement est représentée dans la section Événement Probe , page 83. NOTE : La position sur laquelle l'événement déclencheur a été détecté n'est pas enregistrée.			

Gestion des paramètres et de l'événement d'une étape

Les paramètres et l'événement définis dans une étape sont valides au début de l'exécution de l'étape, par conséquent :

- Si une valeur de paramètre d'étape est modifiée par l'application, la valeur modifiée n'est valide que si la modification est effectuée avant l'activation de l'étape. Le paramètre peut être modifié en utilisant le paramètre de l'objet logiciel alloué dans une POU.
- Une valeur d'objet mémoire (%MW ou %MWx) n'est valide que si elle est mise à jour avant l'activation de l'étape.
- Un événement n'est évalué qu'après l'activation de l'étape. Dans le cas d'un *Probe input event*, si un événement se produit avant l'activation de l'étape, il ne peut pas être détecté.

Gestion des instances de bloc fonctionnel utilisées dans une Motion Task Table

Les instances d'objets logiciels alloués par le système :

- ne peuvent pas être utilisées dans une application POU pour commander un mouvement d'axe.
- Les paramètres de sortie ne sont pas mis à jour par le système durant l'exécution de la Motion Task Table. Cela signifie que les bits de sortie et les paramètres de sortie ne sont pas valides.
- Les paramètres d'entrée :
 - ne peuvent pas être modifiés dans l'éditeur d'instance d'objet logiciel, ni dans l'onglet **Programmation** ;
 - peuvent être utilisés pour modifier dynamiquement la Motion Task Table dans un POU d'application. Pour modifier dynamiquement un paramètre d'entrée dans une instance d'objet logiciel alloué par le système, utilisez l'adresse du paramètre ou le symbole qui lui est associé.

NOTE: L'étape en cours d'exécution peut être modifiée. Par contre, ces modifications seront effectives lors de la prochaine exécution de l'étape seulement.

Exemple de mouvement décrit dans une Motion Task Table :

- Etape : 2
- Type de mouvement : Mouvement relatif
- Objet logiciel : %MC_MOVEREL_PTO1
- Symbole : *Move_Relative_Label2*

Dans l'exemple précédent, le paramètre d'entrée de la vitesse peut être modifié par programmation en utilisant la syntaxe suivante :

- *%MC_MOVEREL_PTO1.Vel*
- *Move_Relative_Label2.Vel*

Gestion des instances de bloc fonction utilisées dans une Motion Task Table :

- Lorsqu'une Motion Task Table est configurée, les instances de bloc fonction réservées sont définies comme **Utilisé**.
- Si toutes les instances d'un bloc fonction particulier sont réservées, le type de mouvement associé ne peut plus être utilisé.

Programmation

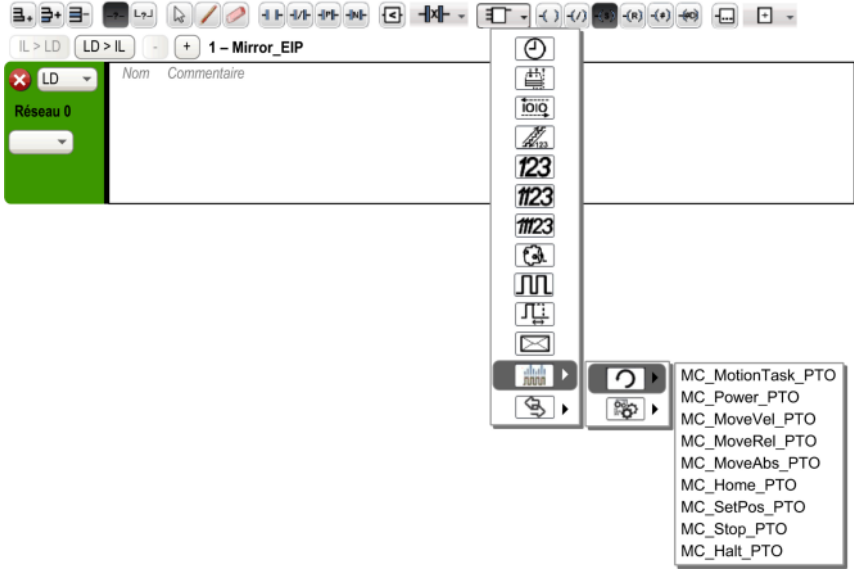
Présentation

Cette section répertorie les blocs fonction utilisés pour programmer la fonction *PTO* et indique comment ajouter ou supprimer ces blocs fonction.

Ajout/suppression d'un bloc fonction

Ajout d'un bloc fonction

Procédez comme suit pour ajouter une instance d'un bloc fonction *PTO* :

Eta-pe	Action
1	Sélectionnez l'onglet Programmation .
2	<p>Sélectionnez Blocs fonction > PTO > Administrative ou Blocs fonction > PTO > Mouvement, comme indiqué dans la figure suivante :</p> 
3	Cliquez sur le réseau pour y placer le bloc fonction sélectionné.
4	Associez les variables d'entrée/sortie du bloc fonction.

NOTE: Configurez les paramètres dans l'onglet **Configuration**.

Pour plus d'informations, reportez-vous au document Modicon M221 Logic Controller - Guide de programmation, Configuration PTO.

Suppression d'un bloc fonction

Procédez comme suit pour supprimer une instance d'un bloc fonction *PTO* :

Eta-pe	Action
1	Dans l'onglet Programmation , cliquez sur l'instance du bloc fonction.
2	Cliquez sur Supprimer pour supprimer le bloc fonction sélectionné.

Blocs fonction PTO

Blocs fonction

La fonction PTO est programmée dans EcoStruxure Machine Expert - Basic à l'aide des blocs fonction suivants :

Catégorie	Bloc fonction	Description
Mouvement (monoaxe), page 111	<i>MC_MotionTask_PTO</i> , page 111	Appelle une Motion Task Table
	<i>MC_Power_PTO</i> , page 114	Active l'alimentation de l'axe, en faisant passer l'axe de l'état <i>Disabled</i> à l'état <i>Standstill</i> . Si le bit <i>%MC_Power_PTO.Status</i> est <i>FALSE</i> , aucun bloc fonction de mouvement ne peut être exécuté pour cet axe.
	<i>MC_MoveVel_PTO</i> , page 117	Provoque le déplacement de l'axe spécifié à la vitesse spécifiée et son passage à l'état <i>Continuous</i> . Ce mouvement continu est maintenu jusqu'à ce que l'une des conditions suivantes soit remplie : une limite logicielle est atteinte, un mouvement d'abandon est déclenché ou une transition vers l'état <i>ErrorStop</i> est détectée.
	<i>MC_MoveRel_PTO</i> , page 120	Déplace l'axe spécifié d'une distance incrémentale à la vitesse spécifiée, et le fait passer à l'état <i>Discrete</i> . La position cible est référencée par rapport à la position en cours au moment de l'exécution, incrémentée d'une distance.
	<i>MC_MoveAbs_PTO</i> , page 123	Provoque le déplacement de l'axe spécifié vers une position donnée à la vitesse spécifiée et son passage à l'état <i>Discrete</i> . L'exécution du bloc fonction se termine avec <i>Error</i> défini sur <i>TRUE</i> si l'axe n'est pas référencé (aucune position de référence absolue définie). Dans ce cas, <i>ErrorId</i> est défini sur <i>InvalidAbsolute</i> .
	<i>MC_Home_PTO</i> , page 126	Commande à l'axe d'exécuter la séquence qui définit la position de référence absolue, et fait passer l'axe à l'état <i>Référencement</i> , page 97. Les détails de cette séquence dépendent des paramètres de configuration <i>Homing</i> .
	<i>MC_SetPos_PTO</i> , page 128	Modifie les coordonnées de l'axe sans aucun mouvement physique.
	<i>MC_Stop_PTO</i> , page 130	Commande un arrêt de mouvement contrôlé et le passage de l'axe à l'état <i>Stopping</i> . Il abandonne l'exécution du mouvement en cours.
	<i>MC_Halt_PTO</i> , page 132	Commande un arrêt de mouvement contrôlé jusqu'à une vitesse nulle, et le passage de l'axe à l'état <i>Discrete</i> . Une fois la sortie <i>Done</i> définie sur <i>TRUE</i> , l'état passe à <i>Standstill</i> .
Administration, page 134	<i>MC_ReadActVel_PTO</i> , page 134	Renvoie la valeur de la vitesse de l'axe.
	<i>MC_ReadActPos_PTO</i> , page 136	Renvoie la valeur de la position de l'axe.
	<i>MC_ReadSts_PTO</i> , page 137	Renvoie l'état de diagramme d'état, page 108 de l'axe.
	<i>MC_ReadMotionState_PTO</i> , page 139	Renvoie l'état de mouvement de l'axe.
	<i>MC_ReadAxisError_PTO</i> , page 140	Renvoie une erreur de contrôle de l'axe, le cas échéant.
	<i>MC_Reset_PTO</i> , page 142	Réinitialise toutes les erreurs liées à l'axe, si les conditions le permettent, pour autoriser une transition entre l'état <i>ErrorStop</i> et l'état <i>Standstill</i> . Cela n'affecte pas la sortie des instances de bloc fonction.
	<i>MC_TouchProbe_PTO</i> , page 143	Active un événement déclencheur sur l'entrée PROBE. L'événement déclencheur permet d'enregistrer la position de l'axe et/ou de lancer un mouvement en mémoire tampon.
	<i>MC_AbortTrigger_PTO</i> , page 145	Annule les blocs fonction qui sont connectés à des événements déclencheurs (par exemple, <i>MC_TouchProbe_PTO</i>).
	<i>MC_ReadPar_PTO</i> , page 146	Récupère les paramètres du bloc fonction PTO.
	<i>MC_WritePar_PTO</i> , page 147	Ecrit les paramètres dans le bloc fonction PTO.

NOTE: Les blocs fonction de mouvement agissent sur la position de l'axe conformément au diagramme d'état du mouvement. Les blocs fonction d'administration n'ont pas d'effet sur l'état du mouvement.

NOTE: Le bloc fonction *MC_Power_PTO*, page 114 est obligatoire avant l'émission d'une commande de mouvement..

⚠ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

- N'utilisez pas la même instance de bloc fonction dans différentes tâches de programme.
- Ne modifiez pas la référence du bloc fonction (AXIS) tant que celui-ci est en cours d'exécution.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Modes de référencement

Présentation

Cette section décrit les modes de référencement de la fonction PTO.

Modes de référencement

Description

Le référencement est la méthode utilisée pour établir le point de référence, ou origine, d'un mouvement absolu.

Un mouvement de référencement peut être effectué selon plusieurs méthodes. Les voies PTO du M221 offrent plusieurs types de mouvement de référencement standard :

- définition de position, page 99,
- référence longue, page 99,
- référence courte avec inversion, page 101,
- référence courte sans inversion, page 100,

Un mouvement de référencement doit être exécuté sans interruption pour que le nouveau point de référence soit valide.

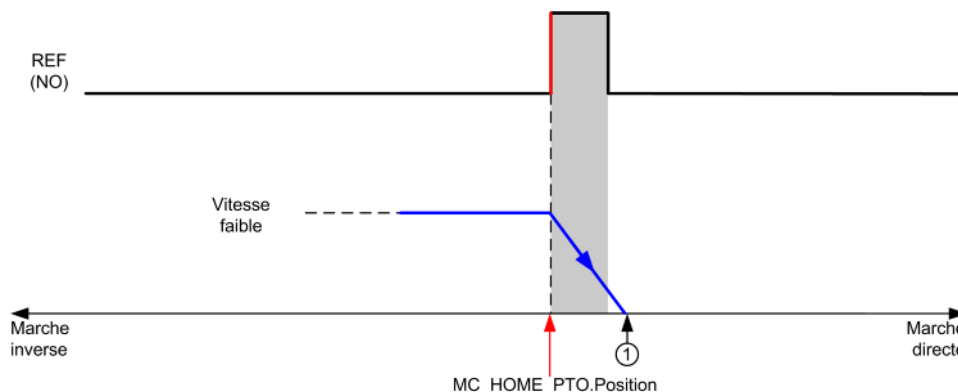
- *%MC_ReadSts_PTO.IsHomed* prend la valeur TRUE lorsqu'un mouvement de référencement a abouti. Si le mouvement de référencement est interrompu, il doit être redémarré.
- *%MC_ReadSts_PTO.IsHomed* prend la valeur FALSE lorsque l'axe est à l'état DISABLED ou lorsqu'aucun mouvement de référencement n'a abouti.

Reportez-vous aux sections *MC_Home_PTO*, page 126 et Codes d'objet de bloc fonction des modes de référencement, page 104.

Position d'origine

Le référencement s'effectue avec un commutateur externe et la position de référencement est définie sur le front du commutateur. Le mouvement est ensuite ralenti jusqu'à l'arrêt.

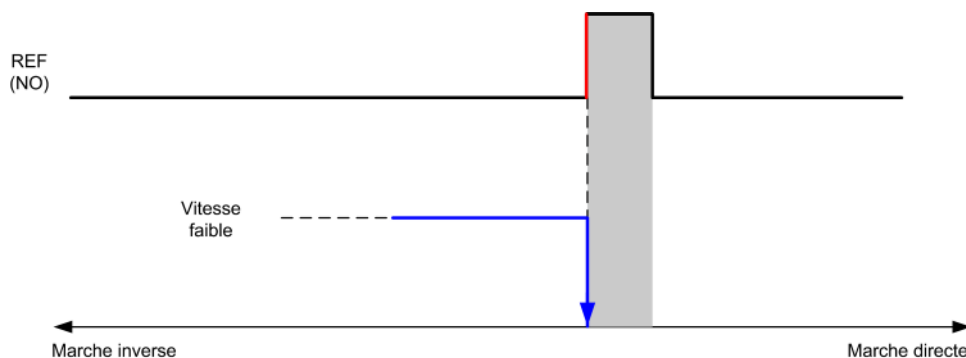
La position réelle de l'axe à la fin de la séquence de mouvement peut donc être différente du paramètre de position défini dans le bloc fonction :



REF (NO) Point de référence (normalement ouvert)

1 Position à la fin du mouvement = $\%MC_HOME_PTO.Position$ + distance "décélération jusqu'à l'arrêt".

Pour simplifier la représentation d'un arrêt dans les diagrammes de modes de référencement, la position réelle de l'axe est présentée ainsi :



REF (NO) Point de référence (normalement ouvert)

Limites

Des limites matérielles sont nécessaires au bon fonctionnement du bloc fonction *MC_Home_PTO* (voir *Limites de positionnement*, page 86 et *MC_Power_PTO*). En fonction du type de mouvement demandé avec le mode de référencement, les limites matérielles garantissent que la fin de course est respectée par le bloc fonction.

Lorsqu'une opération de référencement est lancée dans un sens d'éloignement par rapport au commutateur de référence, les limites matérielles remplissent l'un des deux rôles suivants :

- indiquer qu'une inversion de sens est requise (pour déplacer l'axe vers le commutateur de référence),
- indiquer qu'une erreur a été détectée car la fin de course a été atteinte avant que le commutateur de référence n'ait été rencontré.

Pour les types de mouvement de référencement qui permettent une inversion de sens, l'axe s'arrête selon la décélération configurée lorsque le mouvement atteint la limite matérielle, puis le déplacement reprend en sens inverse.

Pour les mouvements de référencement qui ne permettent pas d'inversion de sens, la procédure de référencement est abandonnée lorsque le mouvement atteint la limite matérielle, une erreur est détectée et l'axe s'arrête conformément à la décélération d'arrêt rapide.

⚠ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

- Assurez-vous que des détecteurs de limite matérielle du contrôleur sont intégrés dans la conception et la logique de l'application.
- Montez les détecteurs de limite matérielle du contrôleur de telle sorte que la distance de freinage soit adéquate.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

NOTE: La distance de freinage appropriée dépend de la vitesse maximum, de la charge (masse) maximum de l'équipement déplacé et de la valeur du paramètre de décélération pour arrêt rapide.

Définition de position

Description

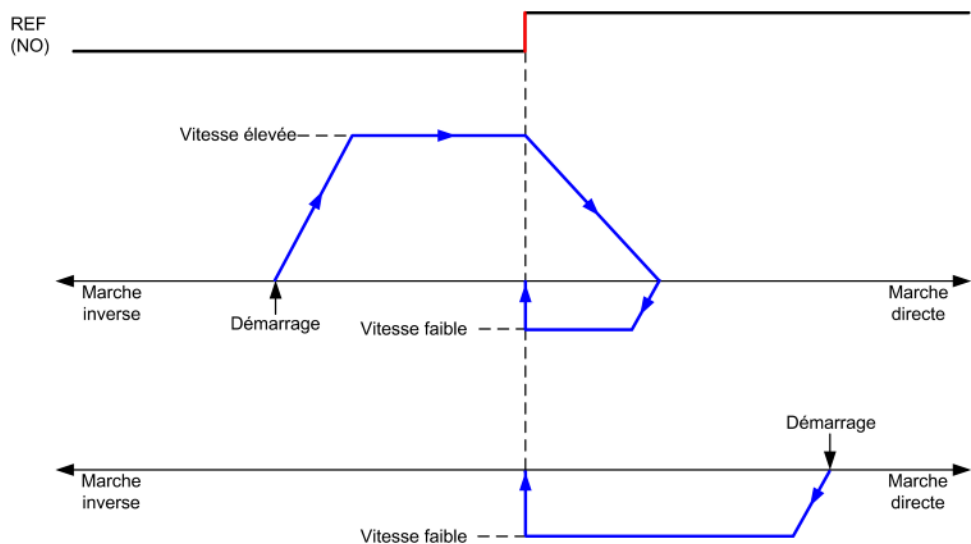
Dans le mode Définition de position, la position actuelle est réglée sur la valeur de position spécifiée. Aucun mouvement n'est effectué.

Référence longue

Référence longue : direction positive

L'origine est définie sur le front descendant du détecteur de référence en direction inverse.

La direction initiale du mouvement dépend de l'état du détecteur de référence :

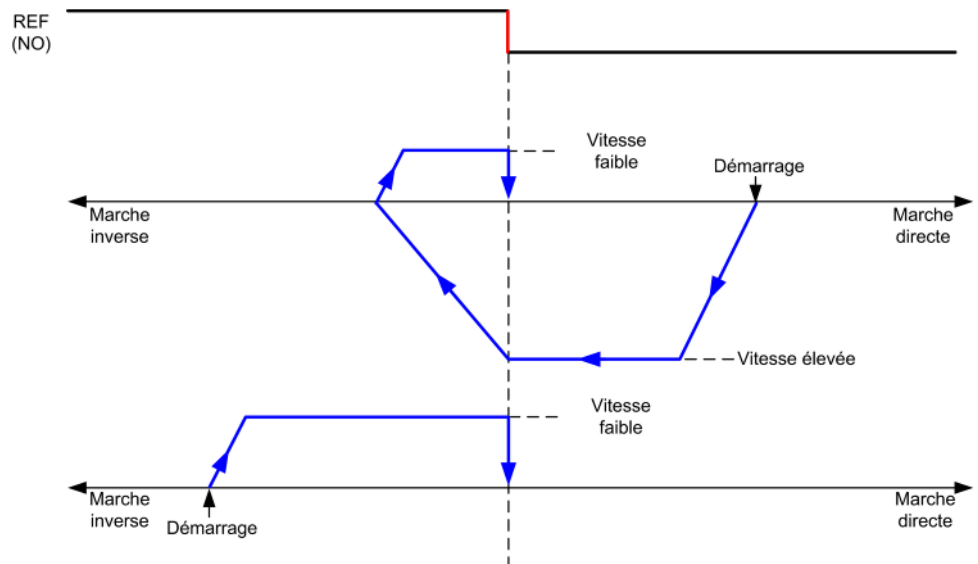


REF (NO) Point de référence (normalement ouvert)

Référence longue : direction négative

L'origine est définie sur le front descendant du détecteur de référence en direction directe.

La direction initiale du mouvement dépend de l'état du détecteur de référence :

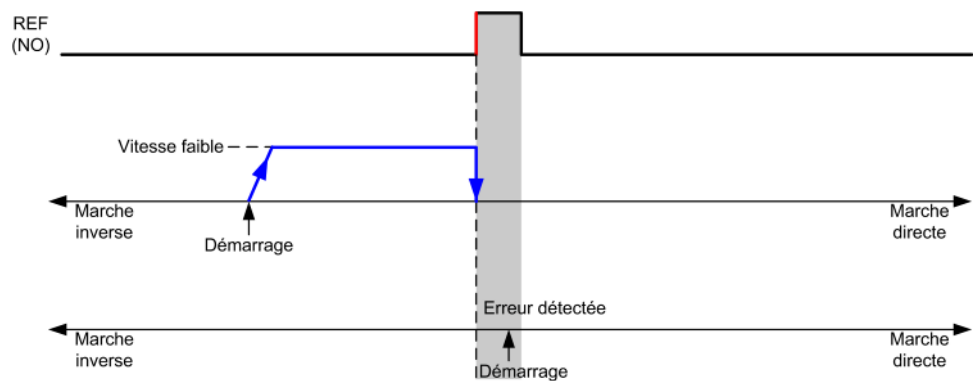


REF (NO) Point de référence (normalement ouvert)

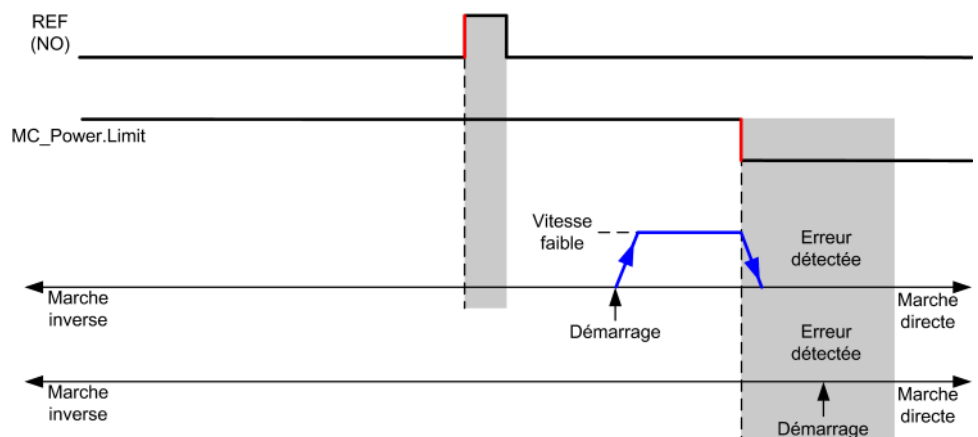
Référence courte sans inversion

Référence courte sans inversion : direction positive

Positionnement à basse vitesse jusqu'au front montant du détecteur de référence en marche directe, sans inversion :



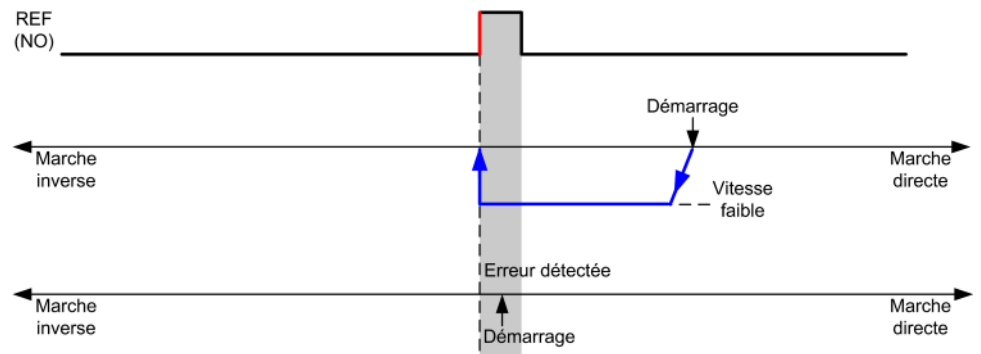
REF (NO) Point de référence (normalement ouvert)



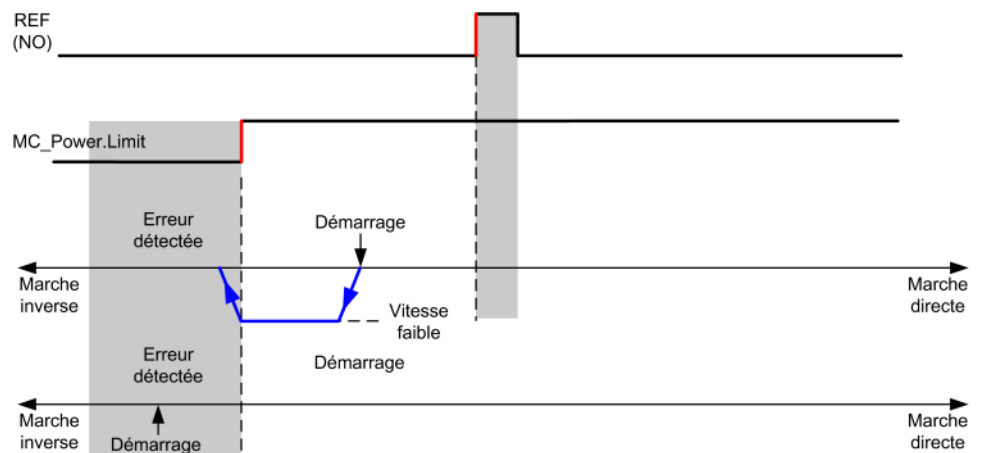
REF (NO) Point de référence (normalement ouvert)

Référence courte sans inversion : direction négative

Positionnement à basse vitesse jusqu'au front descendant du détecteur de référence en marche inverse, sans inversion :



REF (NO) Point de référence (normalement ouvert)



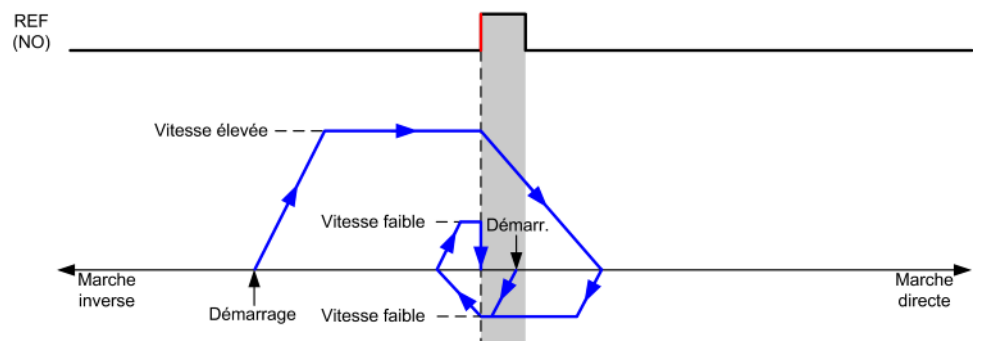
REF (NO) Point de référence (normalement ouvert)

Référence courte avec inversion

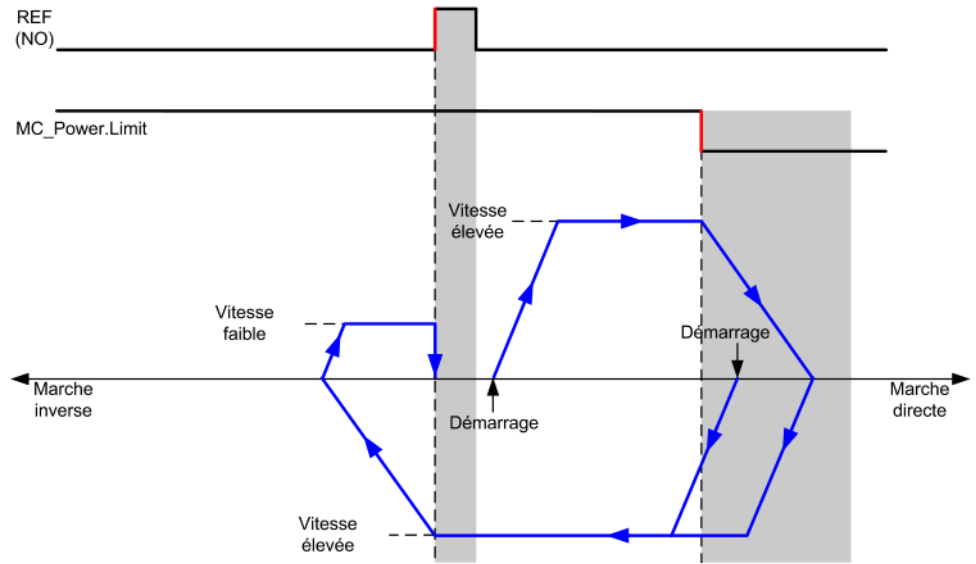
Référence courte avec inversion : direction positive

L'origine est définie sur le front montant du détecteur de référence en direction directe.

La direction initiale du mouvement dépend de l'état du détecteur de référence :



REF (NO) Point de référence (normalement ouvert)

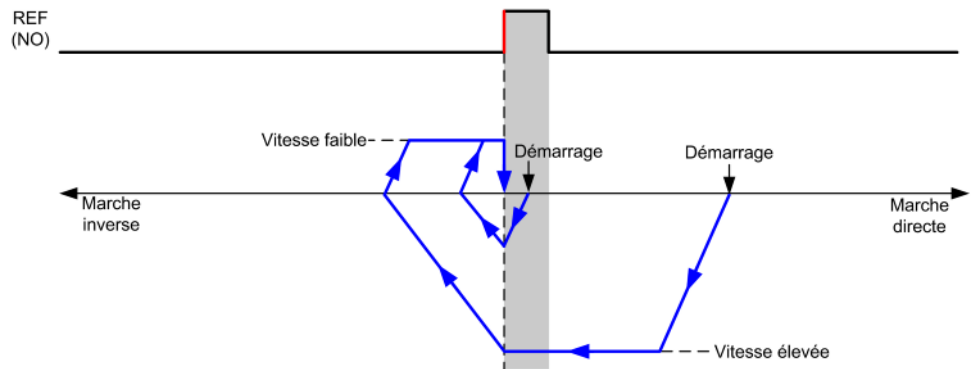


REF (NO) Point de référence (normalement ouvert)

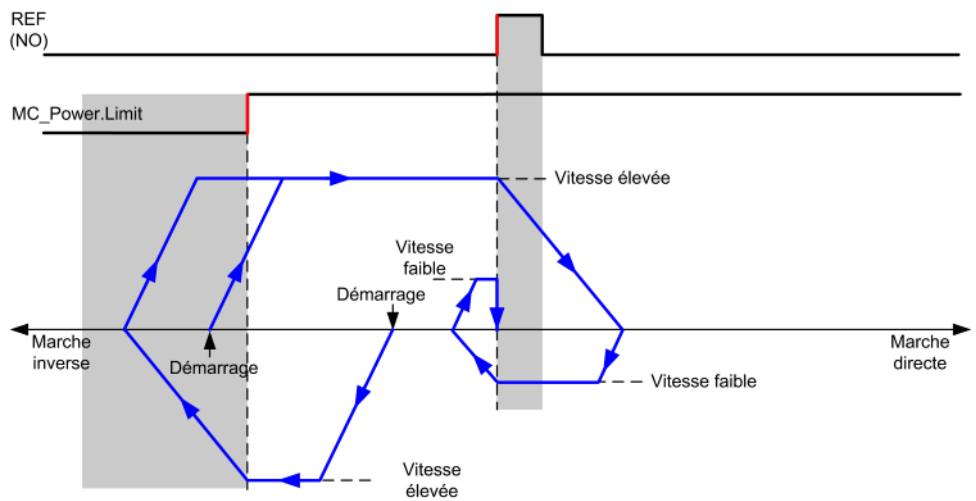
Référence courte avec inversion : direction négative

L'origine est définie sur le front montant du détecteur de référence en direction directe.

La direction initiale du mouvement dépend de l'état du détecteur de référence :



REF (NO) Point de référence (normalement ouvert)



REF (NO) Point de référence (normalement ouvert)

Décalage d'origine

Description

Si l'origine ne peut pas être définie avec assez de précision par les commutateurs, il est possible de faire bouger l'axe jusqu'à une position spécifique décalée par rapport au commutateur d'origine. Le décalage d'origine permet de faire une différence entre origine mécanique et origine électrique.

Le décalage d'origine est défini en nombre d'impulsions (-2 147 483 648...2 147 483 647, valeur par défaut 0). Lorsqu'elle est définie par configuration, la commande *MC_Home_PTO* est exécutée en premier, puis le nombre spécifié d'impulsions est généré à la vitesse basse de référence dans la direction spécifiée.

NOTE: Le temps d'attente entre l'arrêt de la commande *MC_Home_PTO* sur le commutateur d'origine et le début du mouvement de décalage est fixe et réglé sur 500 ms. L'indicateur d'activité de la commande *MC_Home_PTO* n'est libéré qu'une fois le décalage d'origine achevé.

Paramètres des données

Présentation

Cette section décrit les paramètres des données de la fonction *PTO*.

Codes d'objet de bloc fonction

Direction

Ce tableau indique les valeurs des codes d'objet du bloc fonction de direction :

Nom	Valeur	Description
<i>mcPositiveDirection</i>	1	Sens horaire, marche directe, positif (selon le paramètre de configuration Mode de sortie).
<i>mcNegativeDirection</i>	-1	Sens anti-horaire, marche arrière, inverse, négatif (selon le paramètre de configuration Mode de sortie).

Buffer Modes

Ce tableau indique les valeurs des codes d'objet du bloc fonction des modes de tampon :

Nom	Valeur	Description
<i>mcAborting</i>	0	Démarrage immédiat du bloc fonction (mode par défaut). Tout mouvement en cours est abandonné. La file d'attente de mouvements est vidée.
<i>mcBuffered</i>	1	Exécute le bloc fonction dès que le mouvement en cours est terminé (bit <i>Done</i> ou <i>InVel</i> sur TRUE). Il n'y a pas de fusion.
<i>mcBlendingPrevious</i>	3	La vitesse est fusionnée avec la vitesse du premier bloc fonction (fusion avec la vitesse de <i>FB1</i> à la position finale de <i>FB1</i>).
<i>seTrigger</i>	10	Exécute immédiatement le bloc fonction lorsqu'un événement est détecté sur l'entrée Probe. Tout mouvement en cours est abandonné. La file d'attente de mouvements est vidée.
<i>seBufferedDelay</i>	11	Exécute le bloc fonction une fois le mouvement en cours achevé (bit <i>Done</i> ou <i>InVel</i> sur TRUE) et la temporisation écoulee. Il n'y a pas de fusion. Le paramètre <i>Delay</i> est défini à l'aide de <i>MC_WritePar_PTO</i> , avec <i>ParameterNumber</i> 1000.

Modes de référencement

Ce tableau indique les valeurs des codes d'objet du bloc fonction des modes de référencement :

Nom	Valeur	Description
<i>PositionSetting</i>	0	Emplacement.
<i>LongReference</i>	1	Référence longue.
<i>ShortReference_Reversal</i>	20	Référence courte.
<i>ShortReference_NoReversal</i>	21	Référence courte sans inversion.

Paramètre PTO

Ce tableau indique les valeurs des codes d'objet du bloc fonction des paramètres de PTO :

Nom	Numéro du paramètre	R/W	Description
<i>CommandedPosition</i>	1	R	Position commandée.
<i>SWLimitPos (Limite haute)</i>	2	R/W	Limite positive de la position du logiciel.
<i>SWLimitNeg (Limite basse)</i>	3	R/W	Limite négative de la position du logiciel.
<i>EnableLimitPos (Activer les limites de position du logiciel)</i>	4	R/W	Activation du détecteur de limite logicielle positive (0 à 1).
<i>EnableLimitNeg (Activer les limites de position du logiciel)</i>	5	R/W	Activation du détecteur de limite logicielle négative (0 à 1).
<i>MaxVelocityAppl (Vitesse max.)</i>	9	R/W	Vitesse maximale autorisée de l'axe dans l'application (0 à 100 000).
<i>ActualVelocity</i>	10	R	Vitesse de l'axe.
<i>CommandedVelocity</i>	11	R	Vitesse commandée.
<i>MaxAccelerationAppl (Acc. max.)</i>	13	R/W	Accélération maximale autorisée de l'axe dans l'application (0 à 100 000).
<i>MaxDecelerationAppl (Déc. max.)</i>	15	R/W	Décélération maximale autorisée de l'axe dans l'application (0 à 100 000).
Réservé	16 à 999	-	Réservé pour la norme PLCopen.
<i>Delay</i>	1000	R/W	Temps en ms (0 à 65 535) Valeur par défaut : 0
<i>EnableDirPos</i>	1004	R/W	Activation de la direction positive Si la valeur est égale à 0, la direction positive n'est pas autorisée sur l'axe. Un bloc fonction de mouvement entraînant un déplacement dans une direction positive se termine par la détection de l'erreur <i>InvalidDirectionValue</i> (3006). Lorsqu'une nouvelle commande de mouvement dans la direction positive vient interrompre un mouvement négatif, l'erreur n'est détectée qu'à la fin de la décélération du mouvement négatif. Valeur par défaut : 1 NOTE: Un changement de valeur n'est pris en compte qu'à la commande de mouvement suivante ou à l'occurrence suivante de vitesse = 0.
<i>EnableDirNeg</i>	1005	R/W	Activation de la direction négative Si la valeur est égale à 0, la direction négative n'est pas autorisée sur l'axe. Un bloc fonction de mouvement entraînant un déplacement dans une direction négative se termine par la détection de l'erreur <i>InvalidDirectionValue</i> (3006). Lorsqu'une nouvelle commande de mouvement dans la direction négative vient interrompre un mouvement positif, l'erreur n'est détectée qu'à la fin de la décélération du mouvement positif. Valeur par défaut : 1 NOTE: Un changement de valeur n'est pris en compte qu'à la commande de mouvement suivante ou à l'occurrence suivante de vitesse = 0.

Codes d'erreur d'axe PTO

Ce tableau répertorie les valeurs des codes d'erreurs d'axe PTO :

Nom	Valeur	Description
<i>NoError</i>	0	Aucune erreur détectée.
Alertes de contrôle d'axe		
<i>InternalError</i>	1000	Erreur interne du contrôleur de mouvement détectée.
<i>DisabledAxis</i>	1001	Le mouvement n'a pas pu être lancé ou a été abandonné parce que l'axe n'est pas prêt.
<i>HwPositionLimitP</i>	1002	La limite de position positive matérielle <i>limP</i> est dépassée.
<i>HwPositionLimitN</i>	1003	La limite de position négative matérielle <i>limN</i> est dépassée.
<i>SwPositionLimitP</i>	1004	La limite de position positive logicielle est dépassée.
<i>SwPositionLimitN</i>	1005	La limite de position négative logicielle est dépassée.
<i>ApplicationStopped</i>	1006	L'exécution de l'application a été arrêtée (contrôleur à l'état <i>STOPPED</i> ou <i>HALT</i>).
<i>OutputProtection</i>	1007	La protection des sorties contre les courts-circuits est active sur les voies PTO. Consultez la description de <i>%S10</i> et <i>%SW139</i> dans le document Modicon M221 Logic Controller - Guide de programmation, bits système et mots système (voir Modicon M221 Logic Controller - Guide de programmation).
<i>OutputReset</i>	1008	<i>%S9</i> a forcé toutes les sorties à prendre la valeur 0. Voir la section Bits système.
Conseils pour le contrôle d'axe		
<i>WarningVelocityValue</i>	1100	Le paramètre de vitesse commandée étant hors plage, la vitesse est limitée à la vitesse maximale configurée.
<i>WarningAccelerationValue</i>	1101	Le paramètre d'accélération commandée étant hors plage, l'accélération est limitée à l'accélération maximale configurée.
<i>WarningDecelerationValue</i>	1102	Le paramètre de décélération commandée étant hors plage, la décélération est limitée à la décélération maximale configurée.
<i>WarningJerkRatioValue</i>	1103	Le paramètre jerk ratio commandé est limité par l'accélération ou la décélération maximale configurée. Dans ce cas, le paramètre jerk ratio est recalculé pour respecter ces valeurs maximales.

Une **Alerte de contrôle d'axe** fait passer l'axe à l'état **ErrorStop** (*MC_Reset_PTO* est obligatoire pour sortir de l'état **ErrorStop**). L'état d'axe obtenu est indiqué par *MC_ReadSts_PTO* et *MC_ReadAxisError_PTO*.

Codes d'erreur de commande de mouvement PTO

Ce tableau répertorie les valeurs des codes d'erreur de commande de mouvement PTO :

Nom	Valeur	Description
<i>NoError</i>	0	Aucune erreur détectée.
Alertes de conseil d'état de mouvement		
<i>ErrorStopActive</i>	2000	Le mouvement n'a pas pu être lancé ou a été abandonné parce qu'il est interdit par une condition ErrorStop .
<i>StoppingActive</i>	2001	Le mouvement n'a pas pu être lancé parce qu'il est interdit par <i>MC_Stop_PTO</i> qui contrôle l'axe (soit l'axe est en train de s'arrêter, soit l'entrée <i>MC_Stop_PTO.Execute</i> l'entrée est maintenue à TRUE).
<i>InvalidTransition</i>	2002	Transition non autorisée, consultez la section Diagramme d'état de mouvement.
<i>InvalidSetPosition</i>	2003	Impossible d'exécuter <i>MC_SetPos_PTO</i> tant que l'axe est en mouvement.
<i>HomingError</i>	2004	La séquence de référencement ne peut pas commencer sur la came de référence dans ce mode.
<i>InvalidProbeConf</i>	2005	L'entrée Probe doit être configurée.
<i>InvalidHomingConf</i>	2006	L'entrée Ref doit être configurée pour ce mode de référencement.
<i>InvalidAbsolute</i>	2007	Un mouvement absolu ne peut pas être exécuté tant que l'axe n'est pas sur une position d'origine. Une séquence de référencement doit être exécutée au préalable (<i>MC_Home_PTO</i>).
<i>MotionQueueFull</i>	2008	Le mouvement n'a pas pu être placé dans le tampon parce que la file d'attente de mouvements est pleine.
<i>InvalidTransitionMotionTask</i>	2009	La tâche de mouvement et les autres blocs fonction de mouvement liés au même axe ne peuvent pas être exécutés simultanément.
Alertes relatives à la plage		
<i>InvalidAxis</i>	3000	Le bloc fonction n'est pas applicable à l'axe spécifié.
<i>InvalidPositionValue</i>	3001	Le paramètre de position est hors limites ou le paramètre de distance donne une position hors limites.
<i>InvalidVelocityValue</i>	3002	Le paramètre de vitesse est hors limites.
<i>InvalidAccelerationValue</i>	3003	Le paramètre d'accélération est hors limites.
<i>InvalidDecelerationValue</i>	3004	Le paramètre de décélération est hors limites.
<i>InvalidBufferModeValue</i>	3005	Le mode de tampon ne correspond pas à une valeur valide.
<i>InvalidDirectionValue</i>	3006	La direction ne correspond pas à une valeur valide ou la direction n'est pas valide parce qu'une limite de position logicielle ou matérielle est dépassée.
<i>InvalidHomeMode</i>	3007	Le mode de référencement n'est pas applicable.
<i>InvalidParameter</i>	3008	Le numéro de paramètre n'existe pas pour l'axe spécifié.
<i>InvalidParameterValue</i>	3009	Valeur de paramètre hors limites.
<i>ReadOnlyParameter</i>	3010	Paramètre en lecture seule.
<i>InvalidStepMotionTask</i>	3011	Le type d'étape de tâche de mouvement n'est pas défini.

Une **alerte d'état de mouvement** ou une **alerte de relative à la plage** n'affecte pas l'état de l'axe, ni le mouvement en cours d'exécution, ni la file d'attente de mouvements. Dans ce cas, l'erreur est locale au bloc fonction applicable : la sortie *Error* est définie sur TRUE et la sortie de l'objet *ErrorId* prend le code d'erreur de la commande de mouvement PTO appropriée.

Modes de fonctionnement

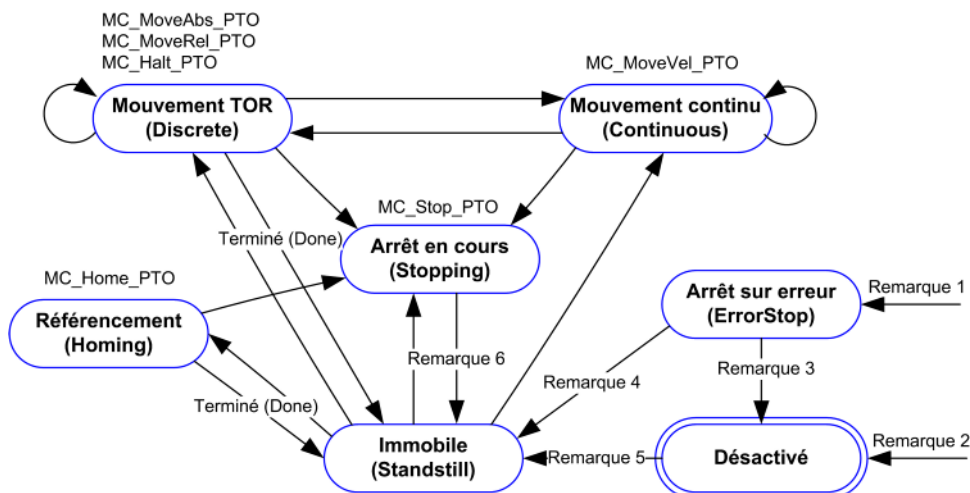
Présentation

Cette section décrit les modes de fonctionnement.

Diagramme d'état de mouvement

Diagramme d'état

L'axe est toujours dans l'un des états définis dans le diagramme suivant :



Remarque 1 Depuis n'importe quel état, lorsqu'une erreur est détectée.

Remarque 2 Depuis n'importe quel état sauf *ErrorStop*, lorsque `%MC_Power_PTO.Status = FALSE`.

Remarque 3 `%MC_Reset_PTO.Done = TRUE` et `%MC_Power_PTO.Status = FALSE`.

Remarque 4 `%MC_Reset_PTO.Done = TRUE` et `%MC_Power_PTO.Status = TRUE`.

Remarque 5 `%MC_Power_PTO.Status = TRUE`.

Remarque 6 `%MC_Stop_PTO.Done = TRUE` et `%MC_Stop_PTO.Execute = FALSE`.

Le tableau suivant décrit les états de l'axe :

Etat	Description
<i>Disabled</i>	Etat initial de l'axe, aucune commande de mouvement n'est autorisée. L'axe n'est pas en position d'origine.
<i>Standstill</i>	Mise sous tension, aucune erreur détectée et aucune commande de mouvement active sur l'axe. La commande de mouvement est autorisée.
<i>ErrorStop</i>	Priorité maximale, applicable en cas d'erreur détectée sur l'axe ou dans le contrôleur. Tout mouvement en cours est annulé par une Décélération d'arrêt rapide . La sortie <i>Error</i> est réglée sur TRUE pour les blocs fonction applicables et la sortie <i>ErrorId</i> indique le code d'erreur. Tant que l'erreur dure, l'état <i>ErrorStop</i> est maintenu. Aucune autre commande de mouvement n'est acceptée jusqu'à la réinitialisation via <i>MC_Reset_PTO</i> .
<i>Homing</i>	Applicable lorsque <i>MC_Home_PTO</i> contrôle l'axe.
<i>Discrete</i>	Applicable lorsque <i>MC_MoveRel_PTO</i> , <i>MC_MoveAbs_PTO</i> ou <i>MC_Halt_PTO</i> contrôle l'axe.

Etat	Description
<i>Continuous</i>	Applicable lorsque <i>MC_MoveVel_PTO</i> contrôle l'axe.
<i>Stopping</i>	Applicable lorsque <i>MC_Stop_PTO</i> contrôle l'axe.

NOTE: les blocs fonction non indiqués dans le diagramme d'état n'affectent pas le changement d'état de l'axe.

La commande de mouvement, y compris les rampes d'accélération et de décélération, ne peut pas dépasser 4 294 967 295 impulsions. A la fréquence maximale de 100 kHz, les rampes d'accélération et de décélération sont limitées à 80 secondes.

Tableau des transitions de mouvement

La voie PTO peut répondre à une nouvelle commande tout en exécutant (et avant de terminer) la commande en cours, comme indiqué dans le tableau ci-après :

Commande		Suivante					
		Home	MoveVel	MoveRel	MoveAbs	Halt	Stop
Intensité	Standstill	Acceptée	Acceptée ⁽¹⁾	Acceptée ⁽¹⁾	Acceptée ⁽¹⁾	Acceptée	Acceptée
	Home	Rejetée	Rejetée	Rejetée	Rejetée	Rejetée	Acceptée
	MoveVel	Rejetée	Acceptée	Acceptée	Acceptée	Acceptée	Acceptée
	MoveRel	Rejetée	Acceptée	Acceptée	Acceptée	Acceptée	Acceptée
	MoveAbs	Rejetée	Acceptée	Acceptée	Acceptée	Acceptée	Acceptée
	Halt	Rejetée	Acceptée	Acceptée	Acceptée	Acceptée	Acceptée
	Stop	Rejetée	Rejetée	Rejetée	Rejetée	Rejetée	Rejetée

⁽¹⁾ Lorsque l'axe est immobile, pour les modes de tampon *mcAborting/mcBuffered/mcBlendingPrevious*, le mouvement débute immédiatement.

Acceptée La nouvelle commande commence à s'exécuter même si la commande précédente n'a pas terminé.

Rejetée La nouvelle commande est ignorée et une erreur est signalée.

NOTE: Quand une erreur est détectée dans la transition de mouvement, l'axe passe à l'état **ErrorStop**. *ErrorId* est réglé sur *InvalidTransition*.

Mode de tampon

Description

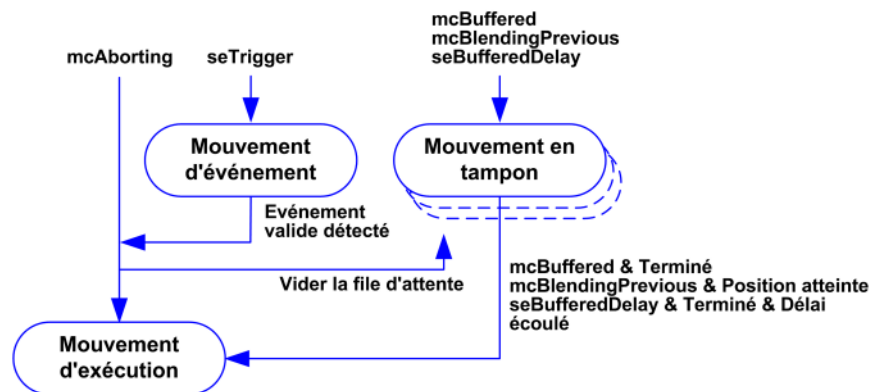
Certains blocs fonction de mouvement ont un objet d'entrée appelé *BufferMode*. Cet objet d'entrée détermine si le bloc fonction s'exécute immédiatement, démarre en cas d'événement Probe ou est placé en mémoire tampon.

Les options disponibles sont définies dans la section Codes d'objet de bloc fonction de modes de tampon, page 104:

- Un mouvement d'abandon (*mcAborting*) démarre immédiatement, annule tout mouvement en cours et vide la file d'attente des mouvements.
- Un mouvement déclenché par un événement (*seTrigger*) est un mouvement d'abandon qui démarre lors d'un événement Probe, page 83.
- Un mouvement en mémoire tampon (*mcBuffered*, *mcBlendingPrevious*, *seBufferedDelay*) est mis en file d'attente, c'est-à-dire ajouté à la suite de tout mouvement en cours d'exécution ou en attente d'exécution, et il démarre lorsque le mouvement précédent est achevé.

Diagramme de la file d'attente de mouvements

La figure suivante illustre la file d'attente de mouvements :



Le tampon ne peut contenir qu'un seul bloc fonction de mouvement.

La condition d'exécution du bloc fonction de mouvement présent dans le tampon est :

- *mcBuffered* : lorsque le mouvement continu en cours est *InVel* ou que le mouvement TOR en cours s'arrête.
- *seBufferedDelay* : lorsque le délai spécifié est écoulé, le mouvement débute à la position actuelle du mouvement continu (*InVel*) ou à la position actuelle du mouvement TOR.
- *mcBlendingPrevious* : lorsque les cibles de position et de vitesse du bloc fonction en cours sont atteintes.

La file d'attente de mouvements est vidée (tous les mouvements en mémoire tampon sont supprimés) :

- Lorsqu'un mouvement d'abandon est déclenché (*mcAborting* ou *seTrigger*) : *CmdAborted* prend la valeur TRUE sur les blocs fonction en mémoire tampon.
- Lorsqu'une fonction *MC_Stop_PTO* est exécutée : La sortie *Error* prend la valeur TRUE sur les blocs fonction en mémoire tampon effacés, avec *ErrorId* = *StoppingActive*.
- Lorsqu'une transition vers l'état **ErrorStop** est détectée : La sortie *Error* prend la valeur TRUE sur les blocs fonction en mémoire tampon, avec *ErrorId* = *ErrorStopActive*.

NOTE:

- Seul un mouvement valide peut être mis en file d'attente. Si l'exécution du bloc fonction se termine avec la sortie *Error* définie sur TRUE, le mouvement n'est pas mis en file d'attente, le mouvement en cours d'exécution se poursuit normalement et la file d'attente n'est pas vidée.
- Lorsque la file d'attente est pleine, la sortie *Error* est définie sur TRUE dans le bloc fonction applicable et la sortie *ErrorId* signale l'erreur *MotionQueueFull*.

Blocs fonction de mouvement

Présentation

Cette section décrit les blocs fonction de **Mouvement**.

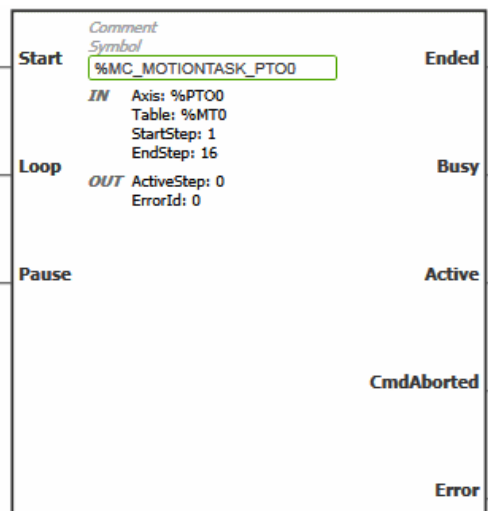
Bloc fonction *MC_MotionTask_PTO*

Description de la fonction

Vous pouvez exécuter à la fois des blocs fonction de mouvement à un seul mouvement et le bloc fonction Motion Task Table (*MC_MotionTask_PTO*) pour un axe.

Toutefois, le bloc fonction *MC_MotionTask_PTO* ne peut pas être exécuté en même temps qu'un autre bloc fonction de mouvement. Sinon, une erreur est détectée et la sortie *ErrorId* est définie sur *InvalidTransitionMotionTask* (2009), page 107.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité et motion task table. Double-cliquez sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et la table cliquez sur **Appliquer**.

Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Valeur initiale	Description
<i>Start</i>	FALSE	<p>Sur le front montant, lance l'exécution du bloc fonction.</p> <p>Les entrées <i>Loop</i> et <i>Pause</i> peuvent être modifiées durant l'exécution du bloc fonction et elles affectent l'exécution en cours.</p> <p>Les valeurs des objets d'entrée <i>Axis</i>, <i>Table</i>, <i>StartStep</i> et <i>EndStep</i> définissent la séquence de mouvement sur un front montant. Les modifications ultérieures de ces objets d'entrée n'affectent pas l'exécution en cours.</p> <p>Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.</p> <p>Si la valeur est FALSE :</p> <ul style="list-style-type: none"> Lors de l'exécution (le mouvement est <i>Busy</i> et <i>Active</i>), les sorties sont actualisées. A l'arrêt de l'exécution, les sorties sont réinitialisées un cycle plus tard.
<i>Loop</i>	FALSE	<p>Si la valeur est TRUE, lorsque l'exécution du bloc fonction s'arrête sans détection d'erreur, la séquence de tâches de mouvement redémarre sur <i>StartStep</i>. La sortie <i>Ended</i> est définie pour un cycle.</p> <p>L'entrée est testée si l'exécution du bloc fonction s'arrête sans détection d'erreur (la valeur de la sortie <i>Ended</i> est TRUE).</p>
<i>Pause</i>	FALSE	<p>Si la valeur est TRUE :</p> <ul style="list-style-type: none"> <i>Active</i> = 1 et <i>Busy</i> = 1 Met l'axe à l'état Halt. <p>Pour atteindre l'état Halt, l'axe décélère à l'état Discrete motion, puis il passe à l'état Standstill si la vitesse = 0.</p> <ul style="list-style-type: none"> L'état Halt est maintenu tant que la valeur de l'entrée <i>Pause</i> est TRUE. Maintient la sortie <i>Active</i> définie même si la vitesse est égale à 0. <p>Si cette entrée est réinitialisée sur FALSE après avoir été définie sur TRUE, l'exécution de la tâche de mouvement est redémarrée dans les conditions suivantes :</p> <ul style="list-style-type: none"> La tâche de mouvement est relancée avec la valeur de la vitesse en cours. Les paramètres de l'étape active sont utilisés. La position absolue de la cible n'est pas modifiée. Si la tâche de mouvement est de type mouvement relatif, aucune distance n'est ajoutée. Dans l'étape, la condition Etape suivante est réinitialisée (par exemple : la temporisation est remise à 0, <i>Probe input event</i> est activé et attend le front configuré).

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	%PTOx	–	Instance d'axe PTO pour laquelle le bloc fonction est exécuté. Le paramètre est défini dans l'instance de bloc fonction atteinte dans l'onglet Programmation > Outils du module. Sélectionnez le paramètre Axe dans la boîte de dialogue Objets PTO > Mouvement > MC_MotionTask_PTO > Propriétés MC_MotionTask_PTO .
<i>Table</i>	%MT	–	Instance de table pour laquelle le bloc fonction est exécuté. Le paramètre est défini dans l'instance de bloc fonction atteinte dans l'onglet Programmation > Outils du module. Sélectionnez le paramètre able dans la boîte de dialogue Objets PTO > Mouvement > MC_MotionTask_PTO > Propriétés MC_MotionTask_PTO .
<i>StartStep</i>	Byte	1	Numéro d'étape qui définit la première étape exécutée dans Motion Task Table. La séquence est exécutée entre <i>StartStep</i> et <i>EndStep</i> . Restriction : $StartStep \leq EndStep$.
<i>EndStep</i>	Byte	16	Numéro d'étape qui définit la dernière étape exécutée dans Motion Task Table. La séquence est exécutée entre <i>StartStep</i> et <i>EndStep</i> . Restriction : $StartStep \leq EndStep$. NOTE: Si la valeur <i>EndStep</i> est supérieure au nombre maximal d'étapes définies dans Motion Task Table, la dernière étape de la table est utilisée.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Ended</i>	0	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur. Comportement des sorties <i>Ended</i> : <ul style="list-style-type: none"> • Si la dernière étape de la séquence de mouvements est un mouvement de type TOR, la sortie agit comme une sortie <i>Done</i>, les autres sorties (<i>Busy</i>, <i>Active</i>, <i>CmdAborted</i> et <i>Error</i>) sont réinitialisées sur 0. • Si la dernière étape de la séquence de mouvements est un mouvement de type continu (vitesse de mouvement), la sortie agit comme une sortie <i>InVel</i>. Comportement des autres sorties : <ul style="list-style-type: none"> ◦ La valeur de <i>Busy</i> et <i>Active</i> est TRUE (1). ◦ La valeur de <i>CmdAborted</i> et <i>Error</i> est FALSE (0) Si une boucle est requise (entrée <i>Loop</i>), la sortie <i>Ended</i> est TRUE pour un cycle de tâches.
<i>Busy</i>	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée.
<i>Active</i>	-	Lorsque la valeur est TRUE, l'instance du bloc fonction contrôle l'axe. Un seul bloc fonction à la fois peut régler la sortie <i>Active</i> sur TRUE pour le même axe.
<i>CmdAborted</i>	-	Si la valeur est TRUE, l'exécution du bloc fonction est arrêtée par une autre commande de mouvement (<i>MC_Stop_PTO</i>) ou la détection d'une erreur.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit les objets de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>ActiveStep</i>	Byte	0	Numéro de l'étape en cours d'exécution dans Motion Task Table.
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Modes de marche

L'exécution d'un Motion Task Table appelé par le bloc fonction *MC MotionTask PTO* est conforme au diagramme des états de mouvement, page 108.

Démarrage de *MC_MotionTask_PTO* : Le bloc fonction ne peut être démarré qu'à partir de l'état **Standstill**.

Arrêt de *MC_MotionTask_PTO* : Le bloc fonction peut être arrêté par l'une des actions suivantes :

- Entrée *Pause* définie sur TRUE.
- Exécution d'un *MC_Stop_PTO*

Comportement des blocs fonction lors de détection d'erreurs :

- Si une erreur de plage ou d'état de mouvement est détectée durant l'exécution du bloc fonction :
 - Une commande d'arrêt de mouvement est appliquée à la tâche de mouvement en utilisant la valeur du paramètre de décélération de l'étape active. Si le paramètre de décélération n'est pas valide, une décélération à arrêt rapide est appliquée.
 - Durant l'arrêt de mouvement contrôlé, les sorties du bloc fonction *Active* et *Busy* restent sur TRUE, avec l'objet de sortie *ActiveStep* = 0.
 - Lorsque le mouvement est arrêté, l'exécution du bloc fonction est terminée avec *Error* = 1, et l'objet de sortie *ErrorId* est défini sur la valeur correspondant au type de l'erreur détectée.
- Si une erreur de contrôle d'axe est détectée, l'axe passe à l'état **ErrorStop**. L'exécution du bloc fonction est terminée avec *Error* = 1 et *ErrorId* = 2000.

Bloc fonction *MC_Power_PTO*

Comportement

L'axe est désactivé lorsque :

- *%MC_Power_PTO.Enable* = FALSE, ou
- *%MC_Power_PTO.DriveReady* = FALSE, ou
- une erreur de limite matérielle est détectée (*HwPositionLimitP* / *HwPositionLimitN*).

Lorsque l'axe est désactivé :

- il passe de l'état *Standstill* à l'état *Disabled*, ou à partir d'un mouvement en cours, il passe à *ErrorStop*, puis à l'état *Disabled* (lorsque l'erreur est réinitialisée).
- *%MC_ReadSts_PTO.IsHomed* est remis à 0 (une nouvelle procédure de référencement devient nécessaire).

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Valeur initiale	Description
<i>Enable</i>	FALSE	Lorsque la valeur est TRUE, le bloc fonction est exécuté. Les valeurs des autres entrées du bloc fonction sont modifiables en permanence, et les sorties sont mises à jour en continu. La valeur FALSE met fin à l'exécution du bloc fonction et réinitialise ses sorties.
<i>DriveReady</i>	FALSE	Signal du variateur, indiquant qu'il est prêt. La valeur doit être TRUE lorsque le variateur est prêt à débiter l'exécution du mouvement. Si le signal du variateur est connecté au contrôleur, utilisez l'entrée appropriée du contrôleur. Si le variateur ne fournit pas ce signal, vous pouvez forcer la valeur TRUE de cette entrée avec n'importe quelle valeur booléenne TRUE.
<i>LimP</i>	TRUE	Information de détection de limite matérielle, dans le sens positif. La valeur est FALSE lorsque la limite matérielle est atteinte. Si le signal de détection de limite matérielle est connecté au contrôleur, utilisez l'entrée appropriée du contrôleur. Si ce signal n'est pas disponible, vous pouvez forcer la valeur TRUE de cette entrée avec n'importe quelle valeur booléenne TRUE.
<i>LimN</i>	TRUE	Information de détection de limite matérielle, dans le sens négatif. La valeur est FALSE lorsque la limite matérielle est atteinte. Si le signal de détection de limite matérielle est connecté au contrôleur, utilisez l'entrée appropriée du contrôleur. Si ce signal n'est pas disponible, vous pouvez forcer la valeur TRUE de cette entrée avec n'importe quelle valeur booléenne TRUE.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.

Sorties

Ce tableau décrit les sorties du bloc fonction :

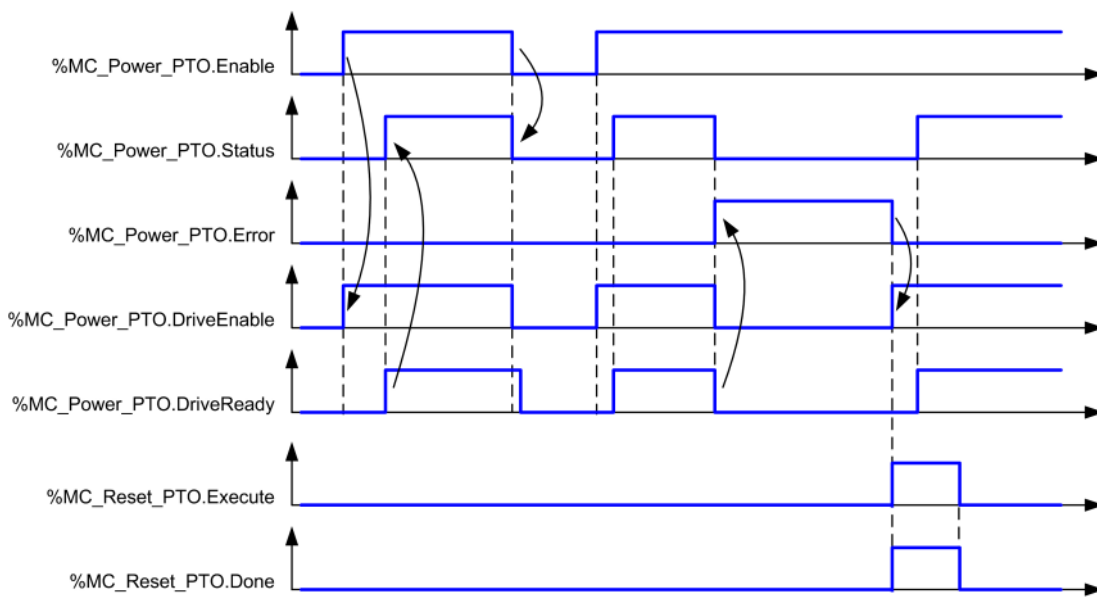
Sortie	Valeur initiale	Description
<i>Status</i>	FALSE	Lorsque la valeur est TRUE, le variateur signale qu'il est prêt à accepter les commandes de mouvement.
<i>DriveEnable</i>	FALSE	Lorsque la valeur est TRUE, cette sortie indique au variateur qu'il peut accepter des commandes de mouvement et qu'il doit donc activer l'alimentation. Si l'entrée du variateur est connectée au contrôleur, utilisez la sortie appropriée du contrôleur. Si le variateur n'a pas d'entrée pour ce signal, laissez la sortie de ce bloc fonction inutilisée.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

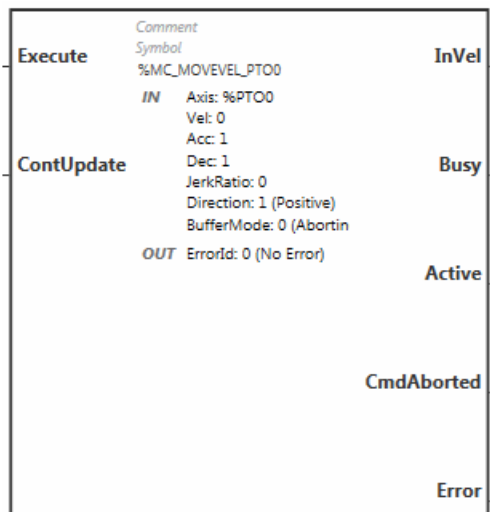
Exemple de chronogramme

Ce schéma détaille le fonctionnement du bloc fonction *MC_Power_PTO* :



Bloc fonction *MC_MoveVel_PTO*

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	<p>Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i>. Une modification postérieure de ces paramètres d'entrée n'a aucun impact sur l'exécution en cours, sauf si l'entrée <i>ContUpdate</i> est TRUE.</p> <p>Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.</p> <p>Si un deuxième front montant est détecté pendant l'exécution du bloc fonction, l'exécution en cours est abandonnée et le bloc fonction est exécuté à nouveau.</p>
<i>ContUpdate</i>	FALSE	<p>Lorsque la valeur est TRUE, le bloc fonction utilise les valeurs modifiées des objets d'entrée (<i>Vel</i>, <i>Acc</i>, <i>Dec</i> et <i>Direction</i>) et les applique à la commande en cours.</p> <p>Pour être prise en compte, cette entrée doit avoir la valeur TRUE avant le front montant sur l'entrée <i>Execute</i>.</p> <p>NOTE: Toute modification de la valeur du paramètre <i>Axis</i> n'est pas prise en compte. Vous devez définir <i>Execute</i> sur 0, puis sur 1 pour modifier <i>Axis</i>.</p>

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>Vel</i>	DINT	0	Vitesse cible. Plage de Hz : 0 à <i>MaxVelocityAppl</i> , page 105
<i>Acc</i>	DINT	0	Accélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxAccelerationAppl</i> , page 105
<i>Dec</i>	DINT	0	Décélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxDecelerationAppl</i> , page 105
<i>JerkRatio</i>	INT	0	Pourcentage d'ajustement de l'accélération/décélération, utilisé pour créer le profil courbé en S, page 82. Plage : 0 à 100
<i>Direction</i>	INT	<i>mcPositiveDirection</i>	Sens du mouvement pour le type CW/CCW de PTO Vers l'avant (CW) = 1 (<i>mcPositiveDirection</i>) Vers l'arrière (CCW) = -1 (<i>mcNegativeDirection</i>)
<i>BufferMode</i>	INT	<i>mcAborting</i>	Mode de transition à partir du mouvement en cours. Consultez la section Tableau des modes de tampon, page 104.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>InVel</i>	FALSE	Lorsque la valeur est TRUE, la vitesse cible est atteinte.
<i>Busy</i>	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée. Le bloc fonction doit être maintenu dans une tâche active du programme d'application tant que la sortie <i>Busy</i> a la valeur TRUE.
<i>Active</i>	-	Lorsque la valeur est TRUE, l'instance du bloc fonction contrôle l'axe. Un seul bloc fonction à la fois peut régler la sortie <i>Active</i> sur TRUE pour le même axe.
<i>CmdAborted</i>	-	Lorsque la valeur est TRUE, l'exécution du bloc fonction est arrêtée par une autre commande de mouvement.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

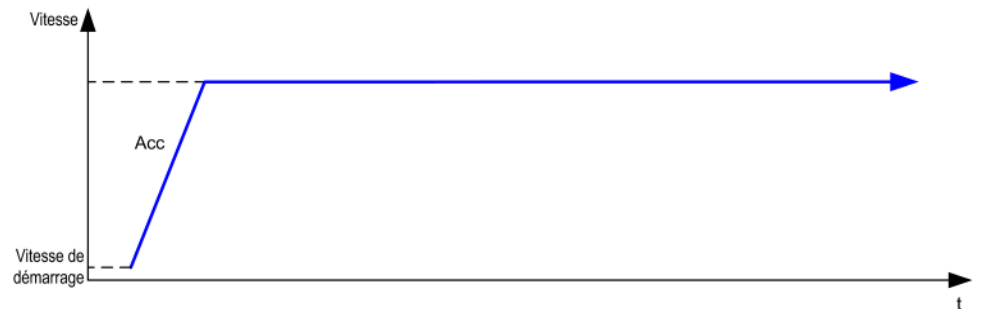
Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

NOTE:

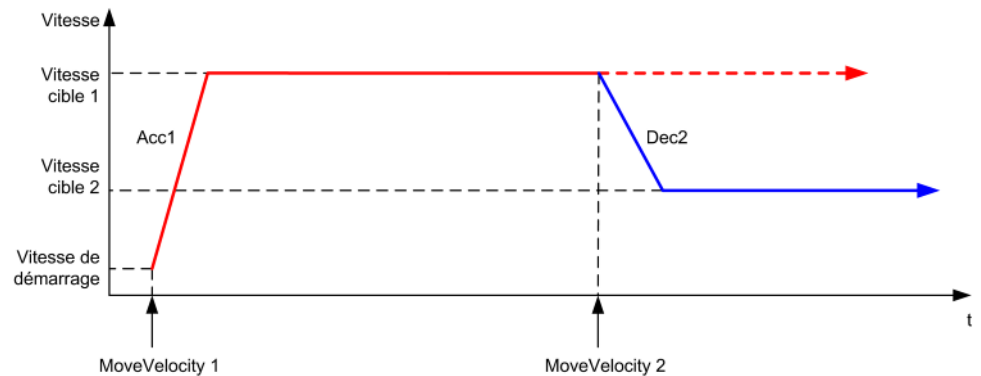
- Pour arrêter le mouvement, le bloc fonction doit être interrompu par un autre bloc fonction qui émet une nouvelle commande.
- Si un mouvement est en cours et que la direction est inversée, ce mouvement est d'abord suspendu avec la décélération du bloc fonction *MC_MoveVel_PTO*, puis le mouvement reprend en sens inverse.
- La durée d'accélération/de décélération du segment ne doit pas excéder 80 s.

Exemple de chronogramme

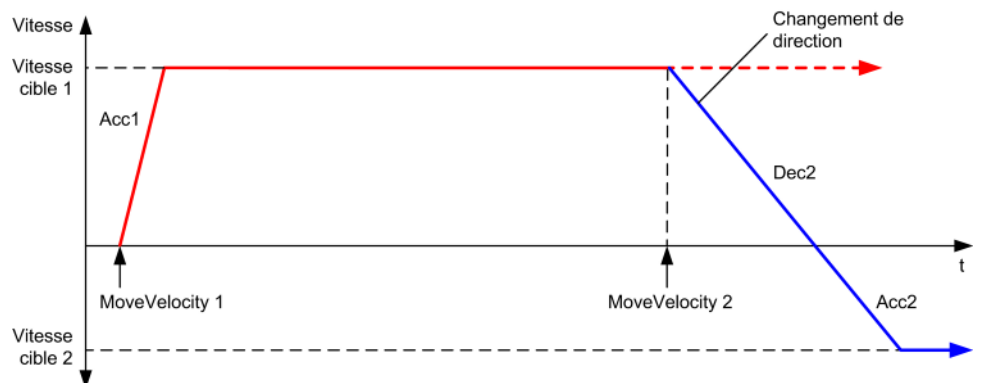
Ce diagramme illustre un profil simple à partir de l'état **Standstill** (immobile) :



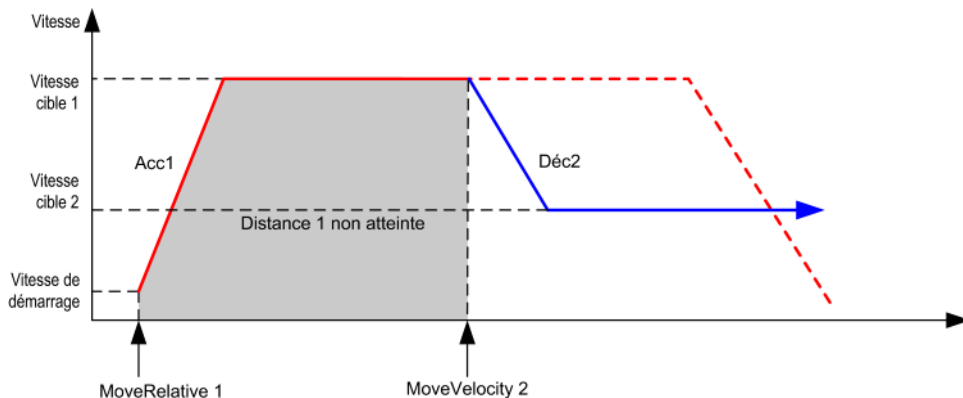
Ce diagramme illustre un profil complexe partir de l'état **Continuous** (continu) :



Ce diagramme illustre un profil complexe à partir de l'état **Continuous** (continu) avec changement de direction :



Ce diagramme illustre un profil complexe partir de l'état **Discrete** (TOR) :



Bloc fonction MC_MoveRel_PTO

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
Execute	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>Distance</i>	DINT	0	Distance relative du mouvement, en impulsions. Le signe indique la direction.
<i>Vel</i>	DINT	0	Vitesse cible. Plage de Hz : 0 à <i>MaxVelocityAppl</i> , page 105
<i>Acc</i>	DINT	0	Accélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxAccelerationAppl</i> , page 105
<i>Dec</i>	DINT	0	Décélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxDecelerationAppl</i> , page 105
<i>JerkRatio</i>	INT	0	Pourcentage d'ajustement de l'accélération/décélération, utilisé pour créer le profil courbé en S, page 82. Plage : 0 à 100
<i>BufferMode</i>	INT	<i>mcAborting</i>	Mode de transition à partir du mouvement en cours. Consultez la section Tableau des modes de tampon, page 104.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur. Lorsqu'un mouvement sur un axe est interrompu par un autre mouvement sur le même axe avant que l'action commandée n'ait abouti, <i>CmdAborted</i> prend la valeur TRUE et <i>Done</i> la valeur FALSE.
<i>Busy</i>	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée. Le bloc fonction doit être maintenu dans une tâche active du programme d'application tant que la sortie <i>Busy</i> a la valeur TRUE.
<i>Active</i>	-	Lorsque la valeur est TRUE, l'instance du bloc fonction contrôle l'axe. Un seul bloc fonction à la fois peut régler la sortie <i>Active</i> sur TRUE pour le même axe.
<i>CmdAborted</i>	-	Lorsque la valeur est TRUE, l'exécution du bloc fonction est arrêtée par une autre commande de mouvement.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

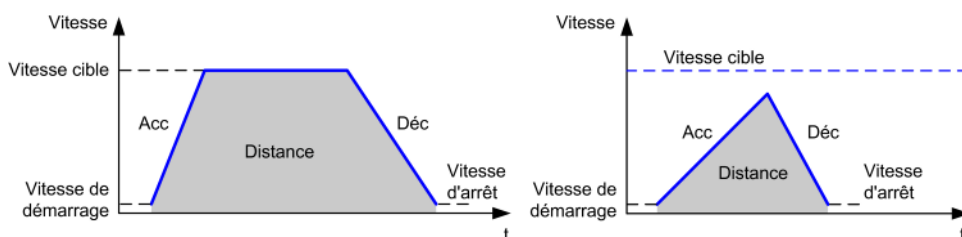
Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

NOTE:

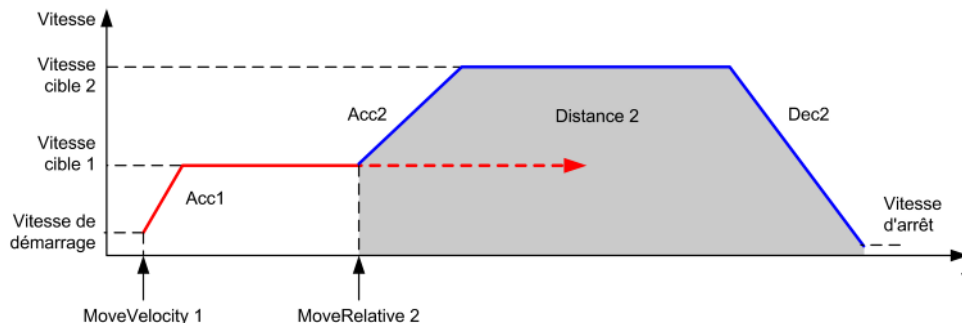
- Le bloc fonction se termine avec la vitesse zéro si aucun autre bloc n'est en attente.
- Si la distance est trop courte pour atteindre la vitesse cible, le profil du mouvement est triangulaire et non trapézoïdal.
- Si un mouvement est en cours et que la distance commandée est dépassée à cause des paramètres du mouvement en cours, l'inversion de direction est gérée automatiquement : le mouvement est tout d'abord suspendu avec la décélération du bloc fonction *MC_MoveRel_PTO*, puis il reprend en sens inverse.
- La durée d'accélération/de décélération du segment ne doit pas excéder 80 s.

Exemple de chronogramme

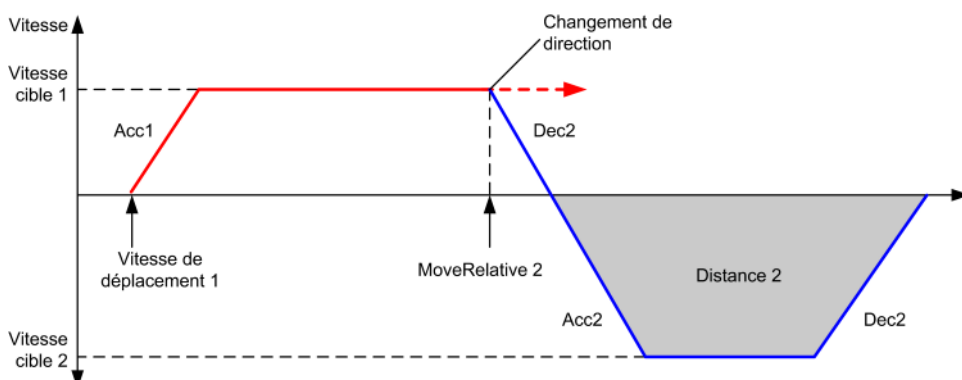
Ce diagramme illustre un profil simple à partir de l'état **Standstill** (immobile) :



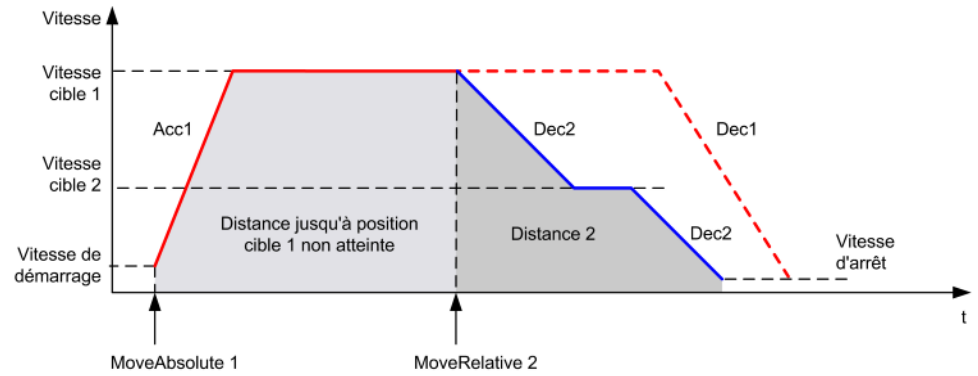
Ce diagramme illustre un profil complexe partir de l'état **Continuoux** (continu) :



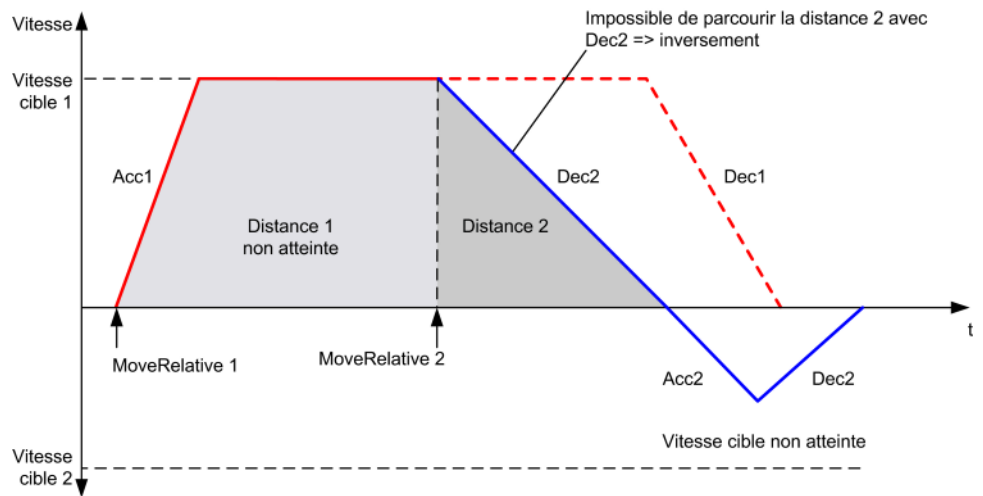
Ce diagramme illustre un profil complexe à partir de l'état **Continuoux** (continu) avec changement de direction :



Ce diagramme illustre un profil complexe partir de l'état **Discrete** (TOR) :

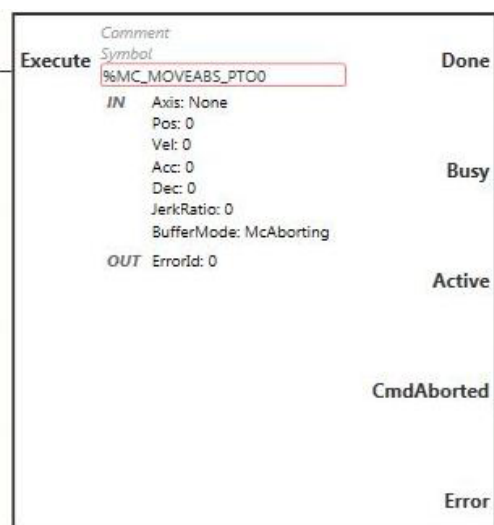


Ce diagramme illustre un profil complexe à partir de l'état **Discrete** (TOR) avec changement de direction :



Bloc fonction *MC_MoveAbs_PTO*

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>Pos</i>	DINT	0	Position de l'axe.
<i>Vel</i>	DINT	0	Vitesse cible. Plage de Hz : 0 à <i>MaxVelocityAppl</i> , page 105
<i>Acc</i>	DINT	0	Accélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxAccelerationAppl</i> , page 105
<i>Dec</i>	DINT	0	Décélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxDecelerationAppl</i> , page 105
<i>JerkRatio</i>	INT	0	Pourcentage d'ajustement de l'accélération/décélération, utilisé pour créer le profil courbé en S, page 82. Plage : 0 à 100
<i>BufferMode</i>	INT	<i>mcAborting</i>	Mode de transition à partir du mouvement en cours. Consultez la section Tableau des modes de tampon, page 104.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur. Lorsqu'un mouvement sur un axe est interrompu par un autre mouvement sur le même axe avant que l'action commandée n'ait abouti, <i>CmdAborted</i> prend la valeur TRUE et <i>Done</i> la valeur FALSE.
<i>Busy</i>	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée. Le bloc fonction doit être maintenu dans une tâche active du programme d'application tant que la sortie <i>Busy</i> a la valeur TRUE.
<i>Active</i>	-	Lorsque la valeur est TRUE, l'instance du bloc fonction contrôle l'axe. Un seul bloc fonction à la fois peut régler la sortie <i>Active</i> sur TRUE pour le même axe.
<i>CmdAborted</i>	-	Lorsque la valeur est TRUE, l'exécution du bloc fonction est arrêtée par une autre commande de mouvement.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

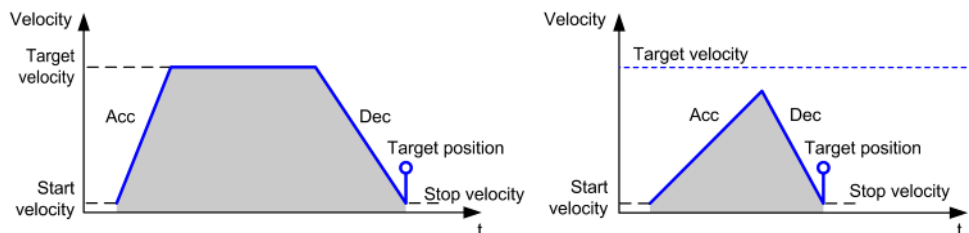
Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

NOTE:

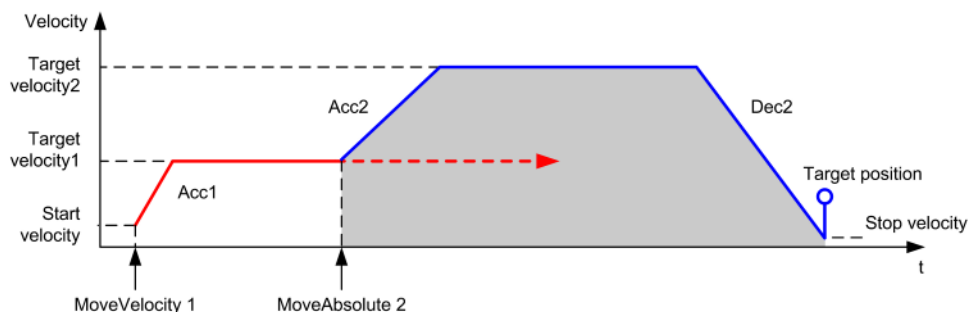
- Le bloc fonction se termine avec la vitesse zéro si aucun autre bloc n'est en attente.
- La direction du mouvement est définie automatiquement en fonction des positions actuelle et cible.
- Si la distance est trop courte pour atteindre la vitesse cible, le profil du mouvement est triangulaire et non trapézoïdal.
- Si la position cible ne peut pas être atteinte dans la direction actuelle, l'inversion est gérée automatiquement. Si un mouvement est en cours, il est d'abord suspendu selon la décélération du bloc fonction *MC_MoveAbsolute_PTO*, puis il reprend en sens inverse.
- La durée d'accélération/de décélération du segment ne doit pas excéder 80 s.

Exemple de chronogramme

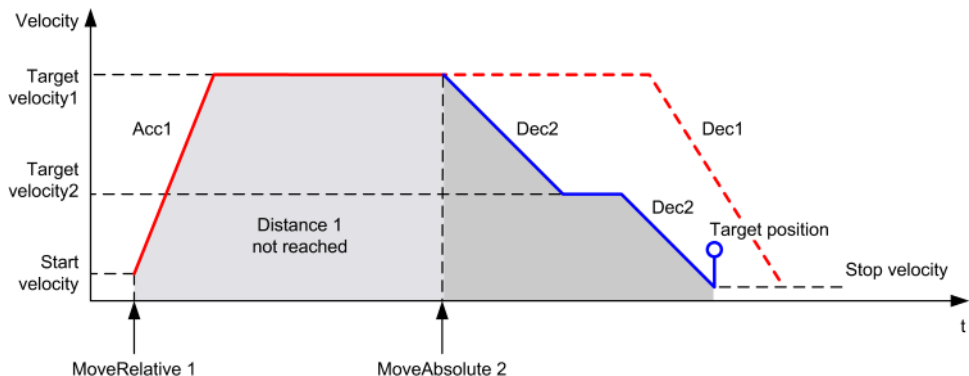
Ce diagramme illustre un profil simple à partir de l'état **Standstill** (immobile) :



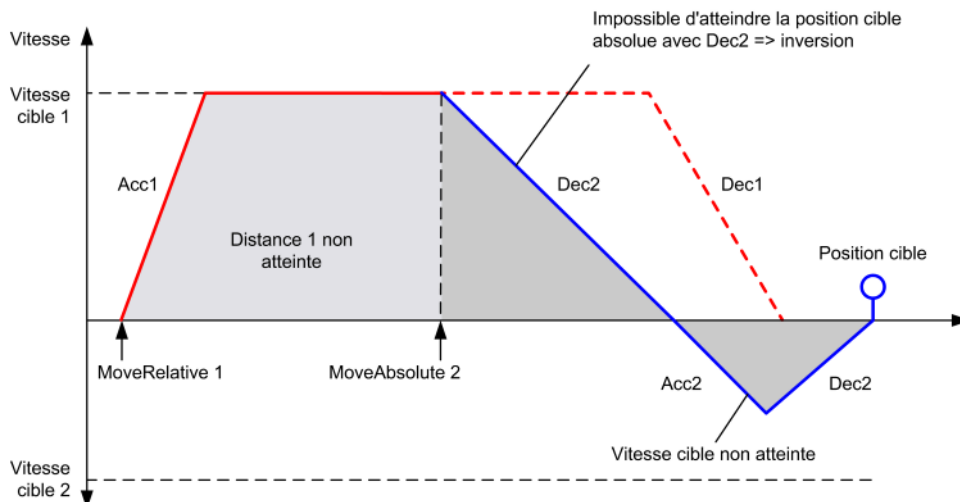
Ce diagramme illustre un profil complexe partir de l'état **Continuous** (continu) :



Ce diagramme illustre un profil complexe partir de l'état **Discrete** (TOR) :

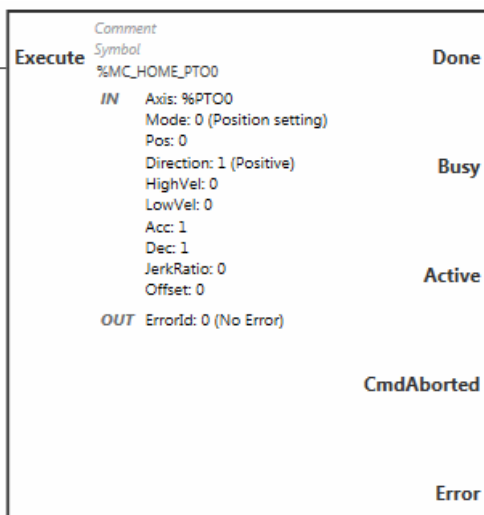


Ce diagramme illustre un profil complexe à partir de l'état **Discrete** (TOR) avec changement de direction :



Bloc fonction MC_Home_PTO

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
Execute	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>Mode</i>	BYTE	0	Type de séquence de référence, page 104 prédéfini.
<i>Pos</i>	DINT	0	Position de l'axe.
<i>HighVel</i>	DINT	0	Vitesse de référencement cible pour la recherche de commutateur de limite ou de référence. Plage de Hz : 1 à <i>MaxVelocityAppl</i> , page 105
<i>LowVel</i>	DINT	0	Vitesse de référencement cible pour la recherche du signal du commutateur de référence. Le mouvement s'arrête lorsque la limite ou la référence est détectée. Plage de Hz : 1 à <i>HighVelocity</i>
<i>Acc</i>	DINT	0	Accélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxAccelerationAppl</i> , page 105
<i>Dec</i>	DINT	0	Décélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxDecelerationAppl</i> , page 105
<i>JerkRatio</i>	INT	0	Pourcentage d'ajustement de l'accélération/décélération, utilisé pour créer le profil courbé en S, page 82. Plage : 0 à 100
<i>Direction</i>	INT	<i>mcPositiveDirection</i>	Sens du mouvement pour le type CW/CCW de PTO Vers l'avant (CW) = 1 (<i>mcPositiveDirection</i>) Vers l'arrière (CCW) = -1 (<i>mcNegativeDirection</i>)
<i>Offset</i>	DINT	0	Distance par rapport au point d'origine. Lorsque le point d'origine est atteint, le mouvement reprend jusqu'à ce que la distance soit couverte. La direction dépend du signe (Décalage d'origine, page 103). Plage : -2 147 483 648 à 2 147 483 647

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur. Lorsqu'un mouvement sur un axe est interrompu par un autre mouvement sur le même axe avant que l'action commandée n'ait abouti, <i>CmdAborted</i> prend la valeur TRUE et <i>Done</i> la valeur FALSE.
<i>Busy</i>	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée. Le bloc fonction doit être maintenu dans une tâche active du programme d'application tant que la sortie <i>Busy</i> a la valeur TRUE.
<i>Active</i>	-	Lorsque la valeur est TRUE, l'instance du bloc fonction contrôle l'axe. Un seul bloc fonction à la fois peut régler la sortie <i>Active</i> sur TRUE pour le même axe.
<i>CmdAborted</i>	-	Lorsque la valeur est TRUE, l'exécution du bloc fonction est arrêtée par une autre commande de mouvement.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

NOTE: La durée d'accélération/de décélération du segment ne doit pas excéder 80 s.

Exemple de chronogramme

Modes de référencement, page 97

Bloc fonction *MC_SetPos_PTO*

Comportement

Ce bloc fonction modifie les coordonnées de la position réelle de l'axe, sans aucun mouvement physique. Il ne s'utilise que lorsque l'axe est dans l'état *Standstill*.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>Pos</i>	DINT	0	Position de l'axe.

Sorties

Ce tableau décrit les sorties du bloc fonction :

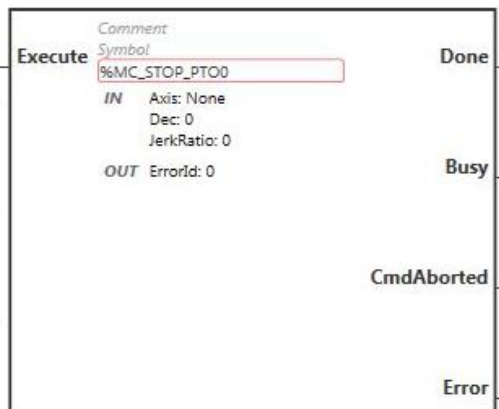
Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_Stop_PTO*

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>Dec</i>	DINT	0	Décélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxDecelerationAppl</i> , page 105
<i>JerkRatio</i>	INT	0	Pourcentage d'ajustement de l'accélération/décélération, utilisé pour créer le profil courbé en S, page 82. Plage : 0 à 100

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur. Lorsqu'un mouvement sur un axe est interrompu par un autre mouvement sur le même axe avant que l'action commandée n'ait abouti, <i>CmdAborted</i> prend la valeur TRUE et <i>Done</i> la valeur FALSE.
<i>Busy</i>	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée. Le bloc fonction doit être maintenu dans une tâche active du programme d'application tant que la sortie <i>Busy</i> a la valeur TRUE.
<i>CmdAborted</i>	-	Lorsque la valeur est TRUE, l'exécution du bloc fonction est arrêtée par une autre commande de mouvement.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

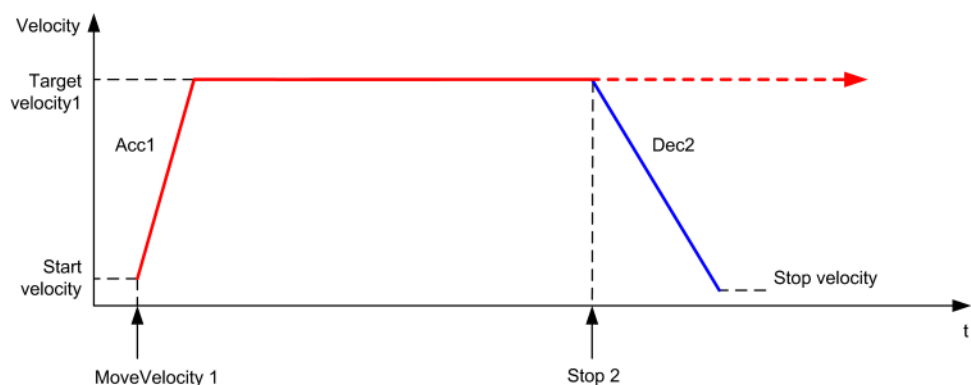
Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

NOTE:

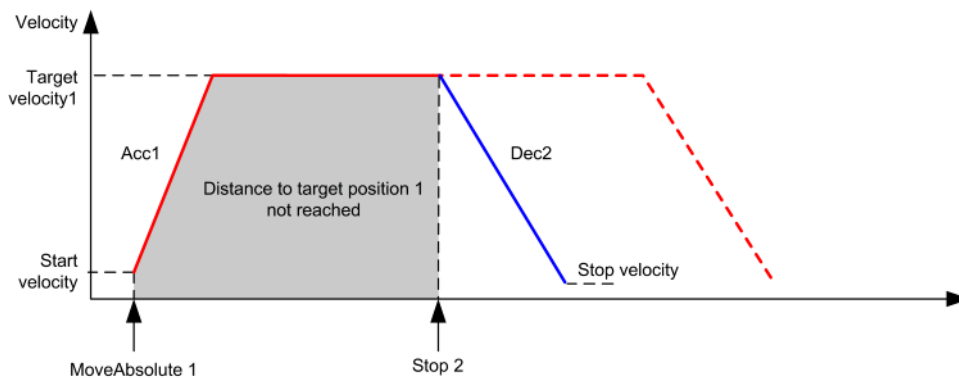
- L'appel de ce bloc fonction dans l'état **Standstill** change l'état en **Stopping**, puis de nouveau en **Standstill** lorsque *Execute* devient FALSE.
- L'état **Stopping** est maintenu tant que la valeur de l'entrée *Execute* est TRUE.
- La sortie *Done* est définie lorsque la rampe d'arrêt est terminée.
- Si *Deceleration* = 0, la décélération d'arrêt rapide est utilisée.
- Le bloc fonction se termine avec la vitesse zéro.
- La durée de décélération du segment ne doit pas excéder 80 s.

Exemple de chronogramme

Ce diagramme illustre un profil simple à partir de l'état **Continuus** (continu) :



Ce diagramme illustre un profil simple à partir de l'état **Discrete** (TOR) :



Bloc fonction *MC_Halt_PTO*

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>Dec</i>	DINT	0	Décélération en Hz/ms Plage (Hz/ms) : 1 à <i>MaxDecelerationAppl</i> , page 105
<i>JerkRatio</i>	INT	0	Pourcentage d'ajustement de l'accélération/décélération, utilisé pour créer le profil courbé en S, page 82. Plage : 0 à 100
<i>BufferMode</i>	INT	<i>mCA-borting</i>	Mode de transition à partir du mouvement en cours. Consultez la section Tableau des modes de tampon, page 104.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur. Lorsqu'un mouvement sur un axe est interrompu par un autre mouvement sur le même axe avant que l'action commandée n'ait abouti, <i>CmdAborted</i> prend la valeur TRUE et <i>Done</i> la valeur FALSE.
<i>Busy</i>	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée. Le bloc fonction doit être maintenu dans une tâche active du programme d'application tant que la sortie <i>Busy</i> a la valeur TRUE.
<i>Active</i>	-	Lorsque la valeur est TRUE, l'instance du bloc fonction contrôle l'axe. Un seul bloc fonction à la fois peut régler la sortie <i>Active</i> sur TRUE pour le même axe.
<i>CmdAborted</i>	-	Lorsque la valeur est TRUE, l'exécution du bloc fonction est arrêtée par une autre commande de mouvement.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

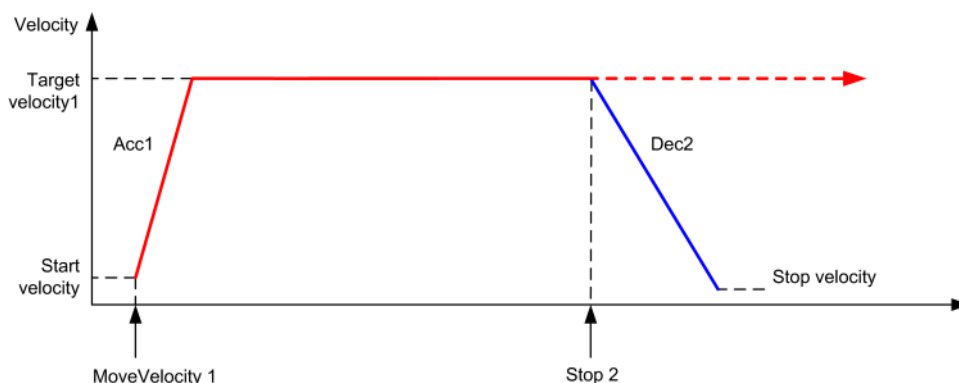
Le tableau suivant décrit l'objet de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

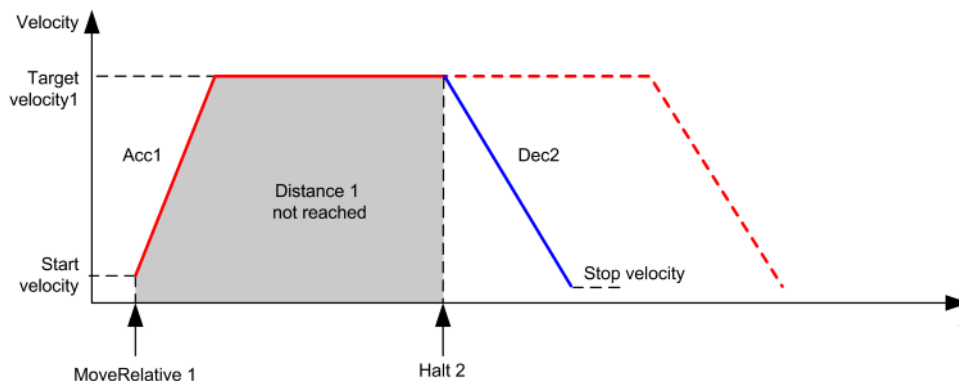
NOTE: Le bloc fonction se termine avec la vitesse zéro.

Exemple de chronogramme

Ce diagramme illustre un profil simple à partir de l'état **Continuous** (continu) :



Ce diagramme illustre un profil simple à partir de l'état **Discrete** (TOR) :



Blocs fonction d'administration

Présentation

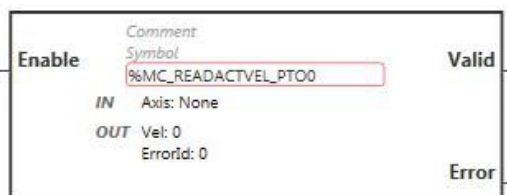
Cette section décrit les blocs fonction d'administration.

Bloc fonction *MC_ReadActVel_PTO*

Description de la fonction

Ce bloc fonction renvoie la valeur de la vitesse réelle de l'axe.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Enable</i>	FALSE	Lorsque la valeur est TRUE, le bloc fonction est exécuté. Les valeurs des autres entrées du bloc fonction sont modifiables en permanence, et les sorties sont mises à jour en continu. La valeur FALSE met fin à l'exécution du bloc fonction et réinitialise ses sorties.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Valid</i>	-	Si la valeur est TRUE, les données des objets du bloc fonction sont valides.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit les objets de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>Vel</i>	DINT	-	Vitesse de l'axe.
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_ReadActPos_PTO*

Description de la fonction

Ce bloc fonction renvoie la valeur de la position réelle de l'axe.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Enable</i>	FALSE	Lorsque la valeur est TRUE, le bloc fonction est exécuté. Les valeurs des autres entrées du bloc fonction sont modifiables en permanence, et les sorties sont mises à jour en continu. La valeur FALSE met fin à l'exécution du bloc fonction et réinitialise ses sorties.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Valid</i>	-	Si la valeur est TRUE, les données des objets du bloc fonction sont valides.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit les objets de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>Pos</i>	DINT	-	Position de l'axe.
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_ReadSts_PTO*

Description de la fonction

Ce bloc fonction renvoie l'état du diagramme des états pour l'axe considéré.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Enable</i>	FALSE	Lorsque la valeur est TRUE, le bloc fonction est exécuté. Les valeurs des autres entrées du bloc fonction sont modifiables en permanence, et les sorties sont mises à jour en continu. La valeur FALSE met fin à l'exécution du bloc fonction et réinitialise ses sorties.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Valid</i>	-	Si la valeur est TRUE, les données des objets du bloc fonction sont valides.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.
<i>IsHomed</i>	FALSE	Lorsque la valeur est TRUE, cette sortie indique que l'axe a été référencé de sorte que le point de référence absolu soit valide et que les commandes de mouvement absolu soient autorisées.
<i>AxisWarning</i>	FALSE	Lorsque la valeur est TRUE, une commande de mouvement a généré une alerte ou un conseil. Utilisez le bloc fonction <i>MC_ReadAxisError_PTO</i> pour obtenir des informations plus détaillées., page 140
<i>QueueFull</i>	FALSE	Lorsque la valeur est TRUE, la file d'attente des mouvements est pleine et aucune autre commande de mouvement n'est autorisée en mémoire tampon.

Le tableau suivant décrit les objets de sortie du bloc fonction :

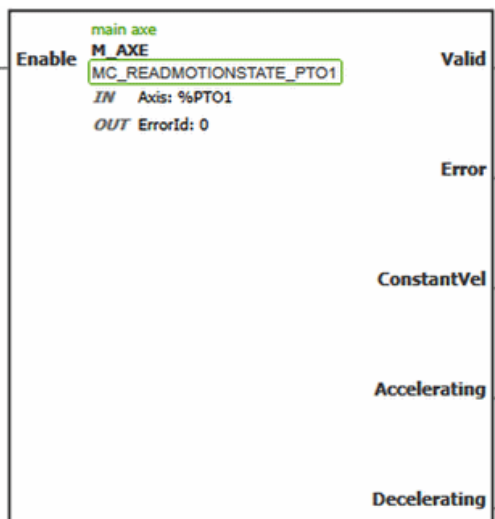
Objet de sortie	Type	Valeur initiale	Description
<i>AxisState</i>	-	-	Code de l'état de l'axe : 0 = axe non configuré 1 = ErrorStop 2 = Disabled (Désactivé) 4 = Stopping (Mise à l'arrêt) 8 = Homing (Référencement) 16 = Standstill (Immobile) 32 = Discrete motion (Mouvement TOR) 64 = Continuous motion (Mouvement continu) Pour plus d'informations, consultez le tableau de description des états, page 108.
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_ReadMotionState_PTO*

Description de la fonction

Ce bloc fonction renvoie l'état de mouvement réel de l'axe.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Enable</i>	FALSE	Lorsque la valeur est TRUE, le bloc fonction est exécuté. Les valeurs des autres entrées du bloc fonction sont modifiables en permanence, et les sorties sont mises à jour en continu. La valeur FALSE met fin à l'exécution du bloc fonction et réinitialise ses sorties.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Valid</i>	-	Si la valeur est TRUE, les données des objets du bloc fonction sont valides.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.
<i>ConstantVel</i>	-	Lorsque la valeur est TRUE, la vitesse de l'axe est constante.
<i>Accelerating</i>	-	Lorsque la valeur est TRUE, la vitesse de l'axe augmente.
<i>Decelerating</i>	-	Lorsque la valeur est TRUE, la vitesse de l'axe diminue.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_ReadAxisError_PTO*

Description de la fonction

Ce bloc fonction lit l'erreur de contrôle d'axe. Si aucune erreur de contrôle d'axe n'est en cours, le bloc fonction renvoie *AxisErrorId* = 0.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Enable</i>	FALSE	Lorsque la valeur est TRUE, le bloc fonction est exécuté. Les valeurs des autres entrées du bloc fonction sont modifiables en permanence, et les sorties sont mises à jour en continu. La valeur FALSE met fin à l'exécution du bloc fonction et réinitialise ses sorties.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Valid</i>	-	Si la valeur est TRUE, les données des objets du bloc fonction sont valides.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit les objets de sortie du bloc fonction :

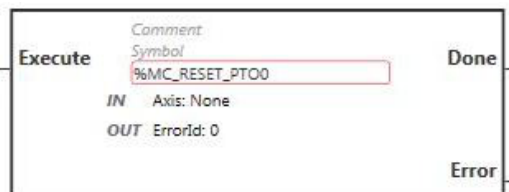
Objet de sortie	Type	Valeur initiale	Description
<i>AxisErrorId</i>	-	-	Codes d'erreur d'axe, valides lorsque la sortie <i>AxisWarning</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur d'axe PTO, page 106.
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_Reset_PTO*

Comportement

Ce bloc fonction réinitialise toutes les erreurs liées à l'axe, si les conditions le permettent, pour autoriser une transition de l'état **ErrorStop** à l'état **Standstill**. Il n'affecte pas la sortie des instances de bloc fonction.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

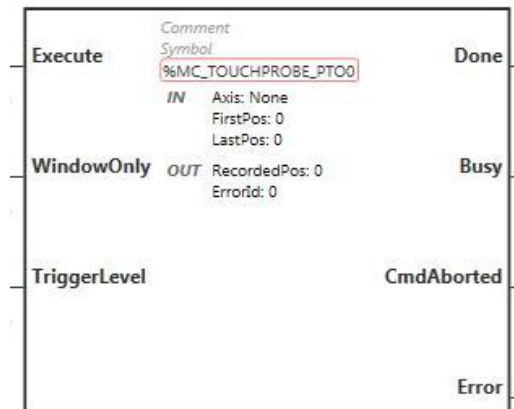
Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_TouchProbe_PTO*

Description de la fonction

Ce bloc fonction est utilisé pour activer un événement déclencheur sur l'entrée de capteur. L'événement déclencheur permet d'enregistrer la position de l'axe et/ou de lancer un mouvement en mémoire tampon.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	<p>Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i>. Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours.</p> <p>Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.</p> <p>Si un deuxième front montant est détecté pendant l'exécution du bloc fonction, l'exécution en cours est abandonnée et le bloc fonction est exécuté à nouveau.</p> <p>Si l'entrée <i>Execute</i> est ensuite réglée sur 0, la position de l'axe est enregistrée et la sortie <i>Done</i> est réglée sur 1 pour un cycle MAST. La position de l'axe est alors réinitialisée et la sortie <i>Done</i> passe à 0.</p>
<i>WindowOnly</i>	FALSE	Lorsque cette sortie à la valeur TRUE, un événement déclencheur n'est reconnu que dans la plage (fenêtre) définie par <i>FirstPosition</i> et <i>LastPosition</i> .
<i>TriggerLevel</i>	FALSE	<p>Lorsque la valeur est TRUE, la position est capturée ou l'événement est déclenché lors du front montant.</p> <p>Lorsque la valeur est FALSE, la position est capturée ou l'événement est déclenché lors du front descendant.</p>

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>FirstPos</i>	DINT	0	Début de la position absolue à partir de laquelle les événements déclencheurs sont acceptés (valeur incluse dans la fenêtre d'activation).
<i>LastPos</i>	DINT	0	Fin de la position absolue à partir de laquelle les événements déclencheurs sont acceptés (valeur incluse dans la fenêtre d'activation).

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur. Lorsqu'un mouvement sur un axe est interrompu par un autre mouvement sur le même axe avant que l'action commandée n'ait abouti, <i>CmdAborted</i> prend la valeur TRUE et <i>Done</i> la valeur FALSE.
<i>Busy</i>	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée. Le bloc fonction doit être maintenu dans une tâche active du programme d'application tant que la sortie <i>Busy</i> a la valeur TRUE.
<i>CmdAborted</i>	-	Lorsque la valeur est TRUE, l'exécution du bloc fonction est arrêtée par une autre commande de mouvement.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit les objets de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>RecordedPos</i>	-	-	Position à laquelle l'événement déclencheur a été détecté.
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

NOTE:

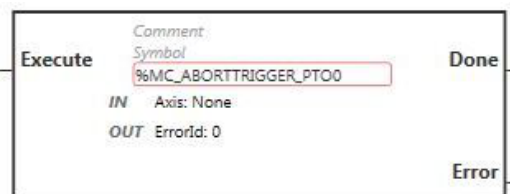
- Seule une instance de ce bloc fonction est autorisée sur le même axe.
- Seul le premier événement après le front montant sur la sortie *MC_TouchProbe_PTO* du bloc fonction *Busy* est valide. Dès que la sortie *Done* est définie sur TRUE, les événements suivants sont ignorés. Le bloc fonction doit être réactivé pour répondre à d'autres événements.

Bloc fonction *MC_AbortTrigger_PTO*

Description de la fonction

Ce bloc fonction est utilisé pour abandonner des blocs fonction qui sont connectés à des événements déclencheurs (par exemple, *MC_TouchProbe_PTO*).

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_ReadPar_PTO*

Description de la fonction

Ce bloc fonction est utilisé pour obtenir des paramètres à partir de la fonction PTO.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Enable</i>	FALSE	Lorsque la valeur est TRUE, le bloc fonction est exécuté. Les valeurs des autres entrées du bloc fonction sont modifiables en permanence, et les sorties sont mises à jour en continu. La valeur FALSE met fin à l'exécution du bloc fonction et réinitialise ses sorties.

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>ParNumber</i>	DINT	0	Code du paramètre que vous souhaitez lire ou écrire. Pour plus d'informations, consultez la section Tableau des paramètres PTO, page 105.

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Valid</i>	-	Si la valeur est TRUE, les données des objets du bloc fonction sont valides.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit les objets de sortie du bloc fonction :

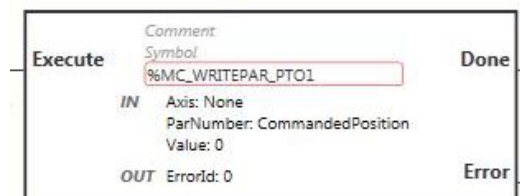
Objet de sortie	Type	Valeur initiale	Description
<i>Value</i>	DINT	0	Valeur du paramètre demandé.
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.

Bloc fonction *MC_WritePar_PTO*

Description de la fonction

Ce bloc fonction est utilisé pour écrire des paramètres dans la fonction PTO.

Représentation graphique



NOTE: Lorsque vous accédez au bloc fonction pour la première fois, vous devez le configurer pour utiliser l'axe souhaité. Cliquez deux fois sur le bloc fonction pour afficher ses propriétés, choisissez l'axe et cliquez sur **Appliquer**.

Entrées

Le tableau suivant décrit l'entrée du bloc fonction :

Entrée	Valeur initiale	Description
<i>Execute</i>	FALSE	Sur le front montant, lance l'exécution du bloc fonction. Les valeurs des autres entrées du bloc fonction contrôlent l'exécution du bloc fonction sur le front montant de <i>Execute</i> . Les modifications ultérieures de ces paramètres d'entrée n'affectent pas l'exécution en cours. Les sorties sont définies à l'arrêt de l'exécution du bloc fonction.

Ce tableau décrit les objets d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
<i>Axis</i>	PTOx	-	Instance pour laquelle le bloc fonction est exécuté. Le nom est déclaré dans la configuration du contrôleur.
<i>ParNumber</i>	DINT	0	Code du paramètre que vous souhaitez lire ou écrire. Pour plus d'informations, consultez la section Tableau des paramètres PTO, page 105.
<i>Value</i>	DINT	0	Valeur à écrire dans le paramètre choisi avec l'objet d'entrée <i>ParNumber</i> .

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
<i>Done</i>	FALSE	Si la valeur est TRUE, l'exécution du bloc fonction est terminée sans détection d'erreur.
<i>Error</i>	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
<i>ErrorId</i>	Word	<i>NoError</i>	Codes d'erreur de commande de mouvement, valides lorsque la sortie <i>Error</i> a la valeur TRUE. Consultez la section Tableau des codes d'erreur de commande de mouvement PTO, page 107.


Générateur de fréquence (%FREQGEN)

Contenu de ce chapitre

Description 149
 Configuration 151

Description

Introduction

Le bloc fonction *FREQGEN* du générateur de fréquence  commande une sortie de signal carré à la fréquence définie.

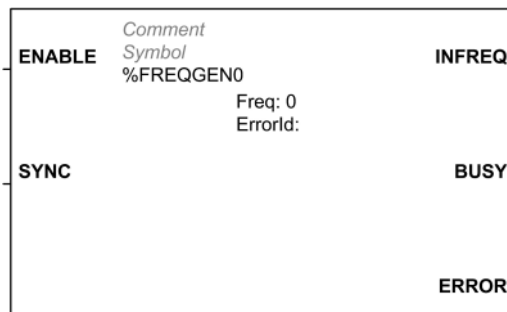
La fréquence est configurable de 0 Hz à 100 kHz, par pas de 1 Hz.

Les caractéristiques de la fonction *FREQGEN* sont présentées ci-après :

Caractéristique	Valeur
Nombre de voies	2 ou 4, selon la référence
Fréquence minimale	0 Hz
Fréquence maximale	100 kHz
Précision sur la fréquence	1 %

Illustration

Cette illustration décrit un bloc fonction *FREQGEN* :



Entrées

Ce tableau décrit les entrées du bloc fonction :

Entrée	Valeur initiale	Description
ENABLE	FALSE	Lorsque la valeur est TRUE, le bloc fonction est exécuté. Les valeurs des autres entrées du bloc fonction sont modifiables en permanence, et les sorties sont mises à jour en continu. La valeur FALSE arrête l'exécution du bloc fonction et réinitialise ses sorties.
SYNC	FALSE	Si un front montant est détecté, la fréquence cible est générée sans attendre la fin de la période en cours.

Le tableau suivant décrit l'objet d'entrée du bloc fonction :

Objet d'entrée	Type	Valeur initiale	Description
Freq	DWORD	-	Fréquence du signal de sortie <i>Frequency Generator</i> en Hz. Définissez la fréquence dans la table des Propriétés des générateurs d'impulsions, page 151 (Plage : minimum 0 (0 Hz) ... maximum 100 000 (100 kHz))

Sorties

Ce tableau décrit les sorties du bloc fonction :

Sortie	Valeur initiale	Description
INFREQ	-	Si la valeur est TRUE, le signal du générateur de fréquence est généré à la fréquence définie dans l'objet d'entrée Freq .
BUSY	-	Lorsque la valeur est TRUE, le bloc fonction est en cours d'exécution. Lorsque la valeur est FALSE, l'exécution du bloc fonction est arrêtée. Le bloc fonction doit être conservé dans une tâche active du programme d'application au moins tant que BUSY a la valeur TRUE.
ERROR	FALSE	La valeur TRUE indique qu'une erreur a été détectée. L'exécution du bloc fonction est terminée.

Le tableau suivant décrit l'objet de sortie du bloc fonction :

Objet de sortie	Type	Valeur initiale	Description
ErrorId	Word	<i>NoError</i>	Codes d'erreur, valides lorsque la sortie ERROR est TRUE. Reportez-vous au tableau des codes d'erreur ErrorId ci-dessous.

Codes d'erreur ErrorId

Le tableau suivant indique les valeurs des codes d'erreur du bloc fonction :

Nom	Valeur	Description
<i>NoError</i>	0	Aucune erreur détectée.
<i>OutputProtection</i>	1007	La sortie d'impulsions a une protection de sortie numérique active. Pour plus d'informations, reportez-vous aux objets système %S10 et %SW139 (voir Modicon M221 Logic Controller - Guide de programmation).
<i>OutputReset</i>	1008	%S9 a forcé toutes les sorties à prendre la valeur 0. Reportez-vous à la section Bits système (voir Modicon M221 Logic Controller - Guide de programmation).
<i>InvalidFrequencyValue</i>	3002	L'objet d'entrée de fréquence Freq se situe hors de la plage autorisée.

Configuration

Présentation

Pour configurer la ressource *Pulse Generator*, reportez-vous à la section Configuration de générateurs d'impulsions (voir Modicon M221 Logic Controller - Guide de programmation).

Pour configurer la ressource *Pulse Generator* en tant que *FREQGEN*, reportez-vous à la section Configuration d'un générateur de fréquence (voir Modicon M221 Logic Controller - Guide de programmation).

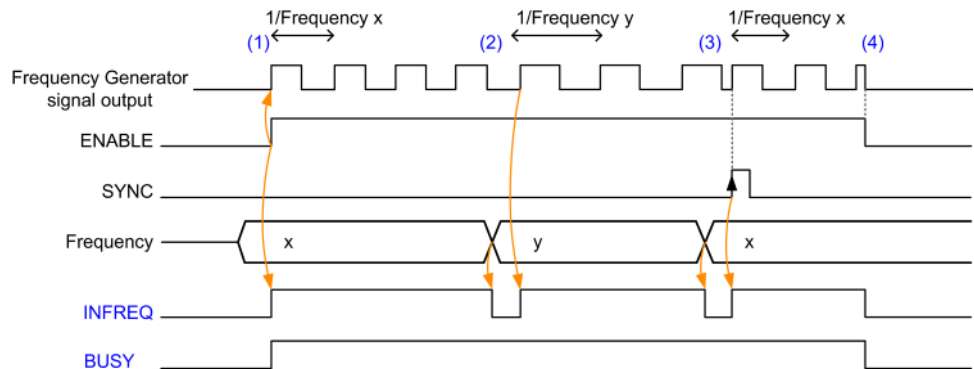
Propriétés

Le bloc fonction *FREQGEN* a les propriétés suivantes :

Propriété	Description	Valeur
Utilisée	Adresse utilisée	Si ce paramètre est sélectionné, l'adresse en question est actuellement utilisée dans un programme.
Adresse	<i>%FREQGENi</i> Adresse du générateur de fréquence	Identificateur de l'instance, <i>i</i> étant compris entre 0 et le nombre d'objets disponibles sur le Logic Controller. Pour connaître le nombre maximum d'objets <i>FREQGEN</i> , consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole	Symbole associé à cet objet. Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert - Basic - Guide d'exploitation).
Freq	Fréquence	Fréquence du signal de sortie du générateur de fréquence, en Hz. Valeur minimale : 0 (0 Hz). Valeur maximale : 100000 (100 kHz). La valeur par défaut est 0.
Commentaire	Commentaire	Un commentaire (facultatif) peut être associé à cet objet. Double-cliquez dans la colonne Commentaire et tapez le commentaire de votre choix.

Chronogramme

Le diagramme suivant indique la chronologie de fonctionnement du bloc fonction *FREQGEN* :



(1) L'entrée *ENABLE* est mise à 1. Le signal du générateur de fréquence est généré sur la sortie dédiée. La sortie *INFREQ* est définie sur 1. La sortie *BUSY* est définie sur 1.

(2) La valeur de fréquence est modifiée. La sortie *INFREQ* est définie sur 0 jusqu'à ce que la nouvelle fréquence soit générée sur la sortie dédiée. La sortie *BUSY* reste sur 1.

(3) L'entrée *SYNC* est mise à 1. Le cycle en cours du générateur de fréquence s'arrête et un nouveau cycle démarre. La sortie *INFREQ* est définie sur 1. La sortie *BUSY* reste sur 1.

(4) L'entrée *ENABLE* est mise à 0. La génération de fréquence s'arrête. La sortie *INFREQ* est définie sur 0. La sortie *BUSY* est définie sur 0.

Lorsque l'application est arrêtée, la génération de fréquence s'arrête sans attendre la fin du cycle de génération d'impulsions. La sortie *Error* reste sur *FALSE*.

Si une erreur est détectée, elle est automatiquement acquittée lorsque la condition de l'erreur est éliminée.

Fonctions logicielles avancées

Contenu de cette partie

Fonction PID.....	154
-------------------	-----

Présentation

Cette section décrit la fonction PID.

Fonction PID

Contenu de ce chapitre

Modes de marche des PID	154
Configuration Auto-Tuning du PID	155
Configuration standard du PID	158
Assistant PID	168
Programmation du PID	177

Modes de marche des PID

Modes de fonctionnement des PID

Introduction

Le contrôleur EcoStruxure Machine Expert - Basic *PID* propose 4 modes de fonctionnement qui sont configurables via l'onglet **General**, page 169 de l'**Assistant PID** dans EcoStruxure Machine Expert - Basic.

Ces modes de fonctionnement *PID* sont les suivants :

- Mode PID
- Mode AT + PID
- Mode AT
- Adresse de mot

Mode PID

Le mode de contrôleur *PID* simple est actif par défaut au démarrage du contrôleur *PID*. Les valeurs de gain K_p , T_i et T_d à spécifier dans l'onglet **PID**, page 172 doivent être connues à l'avance pour contrôler correctement le processus. Vous pouvez choisir le type de correcteur du contrôleur (PID ou PI) dans l'onglet **PID** de l'écran, page 168 **Assistant PID**. Si le type de correcteur PI est sélectionné, le champ **Td** est désactivé.

En mode *PID*, la fonction Auto-Tuning est désactivée et l'onglet **AT**, page 174 de l'écran **Assistant Configuration** n'est donc pas disponible.

Mode AT + PID

Dans ce mode, la fonction Auto-Tuning est active au démarrage du contrôleur *PID*. La fonction Auto-Tuning calcule ensuite les valeurs de gain K_p , T_i et T_d , page 172 et le type d'action PID, page 175. A la fin de la séquence Auto-Tuning, le contrôleur passe en mode *PID* pour la consigne ajustée, en utilisant les paramètres calculés par Auto-Tuning.

Si l'algorithme d'Auto-Tuning détecte une erreur, page 179 :

- Aucun *PID* n'est calculé.
- La sortie d'Auto-Tuning est réglée sur celle appliquée au processus avant le début de l'Auto-Tuning.
- Un message d'erreur apparaît dans la liste déroulante **Liste des états du PID**.
- Le contrôle du *PID* est annulé.

En mode *AT + PID*, la transition de Auto-Tuning à *PID* est automatique et transparente.

Mode AT

Dans ce mode, la fonction Auto-Tuning est active au démarrage du contrôleur *PID* et calcule automatiquement les valeurs de gain K_p , T_i et T_d , page 172 et le type d'action *PID*, page 175. Après la convergence du processus Auto-Tuning et l'aboutissement du processus de détermination des paramètres K_p , T_i et T_d et du type d'action *PID*, page 175 (ou après détection d'une erreur dans l'algorithme Auto-Tuning), la sortie numérique d'Auto-Tuning est définie sur 0 et le message **Auto-Tuning Complete** apparaît dans la liste déroulante des états *PID*, page 179. Le contrôleur *PID* s'arrête alors et attend. Les coefficients *PID* K_p , T_i et T_d calculés sont disponibles dans leurs mots mémoire respectifs ($\%MWx$).

Adresse de mot

Vous sélectionnez ce mode *PID* en affectant la valeur souhaitée à l'adresse de mot associée à cette sélection :

- $\%MWxx = 0$: Le contrôleur est désactivé.
- $\%MWxx = 1$: Le contrôleur fonctionne en mode *PID* simple.
- $\%MWxx = 2$: Le contrôleur fonctionne en mode *AT+ PID*.
- $\%MWxx = 3$: Le contrôleur fonctionne uniquement en mode *AT*.
- $\%MWxx = 4$: Le contrôleur fonctionne en mode *PID* simple, avec le type de correcteur *PI*.

Le mode *Adresse de mot* permet de gérer le mode de fonctionnement du contrôleur *PID* avec l'application et donc de l'adapter à vos besoins.

Configuration Auto-Tuning du PID

Configuration d'Auto-Tuning du PID

Introduction

Cette section vous guide tout au long des étapes nécessaires à la configuration du contrôleur EcoStruxure Machine Expert - Basic *PID* à l'aide de l'Auto-tuning (*AT*).

Cette section contient les étapes suivantes :

Etape	Rubrique
1	Configuration de la voie analogique, page 155
2	Conditions requises pour la configuration du <i>PID</i> , page 156
3	Configuration du <i>PID</i> , page 156
4	Configuration du contrôle, page 157

Etape 1 : Configuration de la voie analogique

Un contrôleur *PID* utilise un signal de retour analogique (appelé valeur de processus) afin de calculer l'algorithme utilisé pour contrôler le processus. Le Logic Controller a une entrée analogique qui permet de mémoriser cette valeur de processus. Pour plus d'informations sur la configuration des entrées analogiques, reportez-vous au document M221 Logic Controller - Guide de programmation.

Si une sortie analogique est utilisée pour piloter le système à contrôler, vérifiez qu'elle est correctement configurée. Vérifiez le module d'extension de sorties analogiques de votre Logic Controller.

Etape 2 : Conditions requises pour la configuration du PID

Avant de configurer le contrôleur *PID*, vérifiez que les phases suivantes ont été effectuées :

Etape	Description
1	PID activé dans le programme, page 177
2	Le mode de scrutation est réglé sur périodique, page 178.

Etape 3 : Configuration du PID

Utilisez une sortie statique avec la fonction PID. L'utilisation d'une sortie relais risque de provoquer rapidement le dépassement des limites du cycle de vie, la rendant ainsi inopérante soit parce que les contacts restent soudés (fermés), soit parce qu'ils se bloquent à l'état ouvert.

⚠ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT OU ÉQUIPEMENT INOPÉRANT

- N'utilisez pas de sorties relais en conjonction avec la fonction PID.
- N'utilisez les sorties statiques que si une sortie numérique est requise pour piloter le système à contrôler.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Pour implémenter un contrôleur *PID* avec Auto-Tuning, procédez comme suit :

Etape	Action
1	Dans l' General tab, page 169 de l'écran Assistant PID (en mode hors ligne), sélectionnez AT+PID (ou AT) ou sélectionnez le paramètre Adresse de mot pour régler le mot associé sur 2 ou 3, dans Operating Modes , page 154.
2	Cochez la case associée à Etats de PID et indiquez l'adresse du mot mémoire dans le champ.
3	Dans l' Input tab, page 171, indiquez l'adresse de l'entrée analogique utilisée comme mesure.
4	Si le paramètre Conversion ou Alarmes est requis, consultez l'onglet Entrée , page 171 de l'écran Assistant PID .
5	Dans l'onglet PID , page 172, indiquez la valeur de la consigne. En règle générale, cette valeur est une adresse mémoire ou une entrée analogique.
6	Le champ Type de correcteur dans l'onglet PID doit être réglé sur PID ou PI .
7	Configurez les paramètres de l'onglet PID : Kp ($\times 0,01$), Ti ($\times 0,1$ s) et Td ($\times 0,1$ s). Lorsque AT+PID ou AT est sélectionné dans Modes de fonctionnement , page 154, les paramètres doivent être des adresses de mot mémoire (%MWxx) pour que l'algorithme Auto-Tuning indique la valeur calculée des paramètres.
8	Renseignez le champ PID Période d'échantillonnage (Ts , page 161) dans l'onglet PID . La Période d'échantillonnage est un paramètre essentiel qui doit être configuré avec précaution.
9	Dans l'onglet AT , le champ Mode AT doit être réglé sur Autoriser par défaut. Indiquez les valeurs Min. et Max. si la case Plage de mesure est cochée (Autoriser). Sélectionnez Correcteur AT dynamique dans la liste contenant Rapide , Moyen , Lent , ou le type de correcteur Adresse de mot . Pour plus d'informations, consultez l'onglet AT de l'Assistant PID, page 168.
10	Dans l'onglet AT , entrez le bit mémoire du Déclencheur AT qui va stocker la valeur du changement d'étape pendant l'Auto-Tuning. Pour plus d'informations, consultez l'onglet AT de l'Assistant PID, page 168.
11	Dans l' Output tab, page 175, définissez l' Action en la sélectionnant dans la liste. Si Adresse de bit est sélectionné, entrez l'adresse de bit mémoire dans le champ Bit . Si nécessaire, vous pouvez configurer les Limites . Si Mode manuel est activé, choisissez un mot mémoire ou une sortie analogique. Si Adresse de bit est sélectionné, entrez un Bit . Pour plus d'informations sur le fonctionnement du mode manuel, reportez-vous à la section Output tab, page 175. Dans le champ Sortie analogique , réglez Output PWM , page 175 sur Autoriser . Définissez une sortie analogique ou un mot mémoire comme adresse du mot. Entrez la valeur dans le champ Période (x0.1 s) et le bit mémoire ou la sortie numérique.
12	Cliquez sur OK pour confirmer la configuration du contrôleur <i>PID</i> .

Etape 4 : Installation du contrôle

Utilisez une sortie statique avec la fonction PID. L'utilisation d'une sortie relais risque de provoquer rapidement le dépassement des limites du cycle de vie, la rendant ainsi inopérante soit parce que les contacts restent soudés (fermés), soit parce qu'ils se bloquent à l'état ouvert.

⚠ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT OU ÉQUIPEMENT INOPÉRANT

- N'utilisez pas de sorties relais en conjonction avec la fonction PID.
- N'utilisez les sorties statiques que si une sortie numérique est requise pour piloter le système à contrôler.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Pour démarrer en mode de fonctionnement, page 154 *AT+PID*, procédez comme suit :

Etape	Action
1	Connectez le PC au contrôleur et transférez l'application.
2	Réglez le contrôleur en mode RUNNING.

NOTE: Avant de régler le contrôleur en mode RUNNING, vérifiez que les conditions de fonctionnement de la machine autorisent ce mode pour le reste de l'application.

Etape	Action
1	Créez une table d'animation contenant les objets définis lors de la configuration. Pour plus d'informations sur la création de tables d'animation, consultez le document <i>Ecostruxure Machine Expert - Basic – Guide d'utilisation</i> .
2	Vérifiez la cohérence de la valeur de processus et des valeurs de l'application. Ce test est important car le bon fonctionnement du contrôleur <i>PID</i> dépend de la précision de la mesure. Si vous avez des doutes concernant la précision de la mesure, mettez le Logic Controller en mode STOP et vérifiez le câblage des voies analogiques. Si l'actionneur n'est pas contrôlé : <ul style="list-style-type: none"> • Pour la sortie analogique, vérifiez la tension ou l'intensité de sortie de la voie analogique. • Pour la sortie PWM, vérifiez : <ul style="list-style-type: none"> ◦ que le voyant de la sortie dédiée est allumé ; ◦ le câblage des alimentations et du circuit 0 V ; ◦ que l'actionneur est sous tension.
3	Dans la table d'animation, vérifiez que : <ul style="list-style-type: none"> • le mode de sortie est automatique ; • tous les paramètres de votre application ont les valeurs appropriées.
4	Définissez la période de scrutation du contrôleur logique de sorte que la Période d'échantillonnage (T_s) du contrôleur <i>PID</i> soit un multiple exact de la période de scrutation. Pour plus d'informations sur la détermination de la période d'échantillonnage, consultez la section <i>Optimisation du PID</i> , page 161.
5	Lorsque la séquence d'Auto-Tuning est terminée, les paramètres Kp , Ti et Td sont stockés dans la RAM du Logic Controller. Les valeurs restent enregistrées tant que l'application est valide (mise hors tension inférieure à 30 jours) et qu'aucun redémarrage à froid n'est effectué.

Le processus Auto-Tuning est répété chaque fois qu'un front montant est détecté sur le bit mémoire **Déclencheur AT**.

NOTE: Lors de l'étalonnage de la fonction Auto-Tuning du PID pour trouver les nouveaux paramètres **Kp**, **Ti** et **Td**, si le contrôle manuel des sorties est activé, il est nécessaire de relancer l'Auto-Tuning de PID à l'issue de ce contrôle pour que les paramètres soient mis à jour.

Configuration standard du PID

Configuration de l'adresse de mot du PID

Introduction

Cette section décrit toutes les étapes nécessaires à la configuration du contrôleur EcoStruxure Machine Expert - Basic *PID* en utilisant le mode de fonctionnement, page 154 **Adresse de mot**. Ce mode offre une plus grande souplesse d'utilisation que les autres modes *PID*.

Cette section contient les étapes suivantes :

Etape	Rubrique
1	Conditions préalables à la configuration du PID, page 159
2	Configuration du PID, page 159
3	Configuration du contrôle, page 160

Etape 1 : Conditions préalables à la configuration du PID

Avant de configurer le *PID*, vérifiez que les phases suivantes ont été effectuées :

Etape	Description
1	Une entrée analogique est configurée ainsi qu'une sortie analogique, le cas échéant. Reportez-vous au document M221 Logic Controller - Guide de programmation.
2	Le PID est activé dans le programme, page 177.
3	Le mode de scrutation est réglé sur périodique, page 178.

Etape 2 : Configuration du PID

Utilisez une sortie statique avec la fonction PID. L'utilisation d'une sortie relais risque de provoquer rapidement le dépassement des limites du cycle de vie, la rendant ainsi inopérante soit parce que les contacts restent soudés (fermés), soit parce qu'ils se bloquent à l'état ouvert.

⚠ AVERTISSEMENT
<p>FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT OU ÉQUIPEMENT INOPÉRANT</p> <ul style="list-style-type: none"> • N'utilisez pas de sorties relais en conjonction avec la fonction PID. • N'utilisez les sorties statiques que si une sortie numérique est requise pour piloter le système à contrôler. <p>Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.</p>

Les étapes suivantes expliquent comment implémenter un contrôleur *PID* en mode **Adresse de mot**. Pour plus d'informations sur la configuration du *PID*, consultez la **section Assistant PID**.

Afin de modifier dynamiquement les paramètres *PID* (en modes hors ligne et en ligne), entrez les adresses de mémoire dans les champs appropriés, afin d'éviter tout basculement en mode hors ligne pour modifier des valeurs à la volée.

Etape	Action
1	Dans l'onglet Général de l'écran Assistant PID (en mode hors ligne), sélectionnez Adresse de mot dans la liste déroulante Modes de fonctionnement . Cochez la case associée à Etats de PID et indiquez l'adresse du mot mémoire dans le champ.
2	Dans l'onglet Entrée , page 171, indiquez l'adresse de l'entrée analogique utilisée comme mesure. Si le paramètre Conversion ou Alarmes est requis, consultez l'onglet Entrée , page 171 de l'Assistant PID, page 168.
3	Dans l'onglet PID , indiquez la valeur de la consigne . En règle générale, cette valeur est une adresse mémoire ou une entrée analogique. Les Paramètres (Kp, Ti et Td) doivent être des adresses de mot mémoire (%MWxx) . Renseignez le champ PID Période d'échantillonnage (Ts, page 173) dans l'onglet PID , page 172. Ce paramètre peut également être un mot mémoire (dont la valeur est configurable à l'aide de la table d'animation). En mode Adresse de mot , le champ Type de correcteur affiche Auto et est grisé (non modifiable manuellement).
4	Dans l'onglet AT , le mode AT doit être défini sur Autoriser . Renseignez les champs Correcteur dynamique et Déclencheur AT . Pour plus d'informations, consultez l'onglet AT , page 174 dans l'écran Assistant PID .
5	Dans l'onglet Sortie , le champ Action doit afficher Adresse de bit . Entrez une adresse de bit mémoire . Le cas échéant, vous pouvez configurer les limites dans le champ Limites de l'onglet Sortie , page 175. Dans le champ Sortie analogique , indiquez l'adresse du mot : une sortie analogique ou un mot mémoire. Le cas échéant, choisissez une option dans le champ PWM sortie . Pour plus d'informations, consultez l'onglet Sortie , page 175 de l'Assistant PID, page 168.
6	Cliquez sur OK pour confirmer la configuration du contrôleur PID .

Etape 3 : Vérification de la configuration

Etape	Action
1	Connectez le PC au Logic Controller et transférez l'application.
2	Réglez le Logic Controller en mode RUNNING.

NOTE: Avant de régler le Logic Controller en mode RUNNING, vérifiez que les conditions de fonctionnement de la machine autorisent ce mode pour le reste de l'application. La procédure reste identique à celle utilisée dans les modes de fonctionnement **AT** et **AT+PID**. La configuration Adresse de mot vous permet de modifier les modes de fonctionnement du PID par voie logicielle. En mode PID, la procédure est beaucoup plus simple, dès lors que les paramètres (Kp, Ti, Td et Ts) sont connus et qu'aucun Auto-Tuning n'est nécessaire.

Le tableau suivant détaille la procédure générique permettant de configurer le contrôleur *PID*.

Etape	Action
1	Créez une table d'animation contenant les objets définis lors de la configuration. Pour plus d'informations, consultez le document <i>EcoStruxure Machine Expert - Basic – Guide d'utilisation</i> .
2	Vérifiez la cohérence de la valeur de processus et des autres valeurs dans la table d'animation. Si vous avez des doutes concernant la précision de la mesure, mettez le Logic Controller en mode STOP et vérifiez le câblage des voies analogiques. Si vous constatez que l'actionneur n'est pas contrôlé : <ul style="list-style-type: none"> • Pour la sortie analogique, vérifiez la tension ou l'intensité de sortie de la voie analogique. • Pour la sortie PWM, vérifiez : <ul style="list-style-type: none"> ◦ que le voyant de la sortie dédiée est allumé ; ◦ le câblage des alimentations et du circuit 0 V ; ◦ que l'actionneur est sous tension.
3	Définissez la période de scrutation du contrôleur logique de sorte que la Période d'échantillonnage (Ts) du contrôleur <i>PID</i> soit un multiple exact de la période de scrutation. Pour plus d'informations sur la période d'échantillonnage , reportez-vous à la section Détermination de la période d'échantillonnage, page 166.
4	Si vous prévoyez d'utiliser la fonction Auto-Tuning, page 161, il se peut que vous deviez activer le Mode manuel, page 165 pour connaître le Correcteur dynamique et le Déclencheur AT définis dans l'onglet AT , page 174 de l' Assistant PID .
5	Mettez le contrôleur de boucle sous tension à l'aide de la table d'animation : <ul style="list-style-type: none"> • Configurez le mode de fonctionnement, page 154. • Activez le contrôleur de PID, page 178. • Modifiez les valeurs définies lors de la configuration, page 159, selon le mode de fonctionnement sélectionné.

Réglage du PID avec la fonction Auto-Tuning (AT)

Introduction

Le mode Auto-Tuning permet de régler automatiquement Kp, Ti, Td et les paramètres d'action pour affiner la convergence de la fonction *PID*. La fonction Auto-Tuning fournie par EcoStruxure Machine Expert - Basic est particulièrement adaptée au réglage automatique des processus thermiques.

Cette section contient les rubriques suivantes :

- Conditions requises par la fonction Auto-Tuning
- Description du processus Auto-Tuning
- Stockage des coefficients calculés
- Réglage des paramètres *PID*
- Lancement du processus Auto-Tuning
- Limites à l'utilisation de l'auto-tuning et du contrôle du *PID*

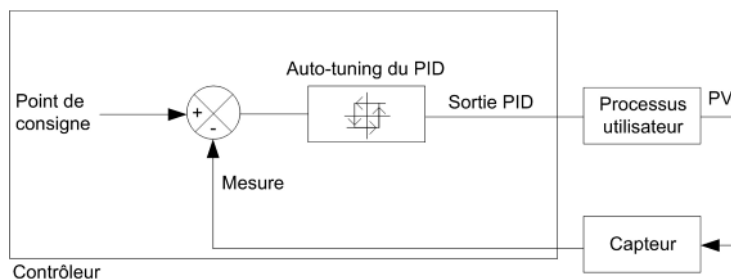
Conditions requises par la fonction Auto-Tuning

Lorsque vous utilisez la fonction Auto-Tuning, vérifiez que le processus de contrôle et le contrôleur logique remplissent les conditions suivantes :

- Conditions requises par le processus :
 - Le processus doit être un système stable en boucle ouverte.
 - Le processus doit être principalement linéaire sur toute la plage de fonctionnement.
 - La réponse du processus à une modification du niveau de la sortie analogique doit suivre un schéma transitoire asymptotique.
 - Le processus est prêt, avec une entrée nulle au début de la séquence Auto-Tuning.
 - Le processus doit être intégralement exempt de perturbations. Sinon, les paramètres calculés seront incorrects ou le processus Auto-Tuning ne fonctionnera pas correctement.
- Configuration requise :
 - Configurez le contrôleur logique en mode de scrutation périodique pour garantir la bonne exécution de la fonction Auto-Tuning.
 - N'utilisez la fonction Auto-Tuning que si aucun autre contrôleur *PID* n'est en cours d'exécution.
 - Configurez les coefficients K_p , T_i et T_d comme des adresses de mot mémoire (%MWxx).
 - Dans l'onglet **Sortie**, définissez le type d'action sur une adresse de bit mémoire (%Mxx).

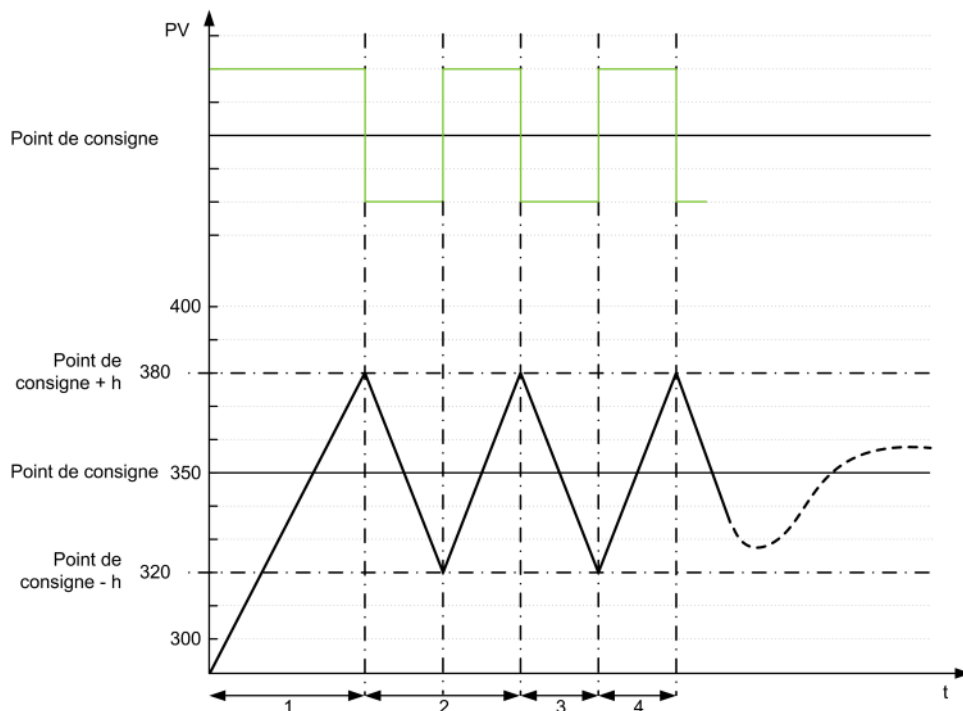
Description du processus Auto-Tuning

L'illustration suivante décrit l'auto-tuning dans le contrôleur et dans l'application :



Description du processus d'étalonnage de l'Auto-Tuning

Le processus d'étalonnage de l'Auto-Tuning se divise en quatre phases consécutives. Chaque phase doit être finalisée pour réussir l'Auto-Tuning. Les courbes de réponse du processus et le tableau suivants décrivent les quatre phases de la fonction d'Auto-Tuning du EcoStruxure Machine Expert - Basic PID :



PV Valeur de processus

█ Sortie PID

h = 1 % (valeur **Max** - valeur **Min**) de **Plage de mesure** dans l'onglet **AT**

---- PID actif

1 à 4 Phases de l'Auto-Tuning (voir ci-dessous)

Le tableau ci-dessous décrit les phases de l'Auto-Tuning :

Phase d'Auto-Tuning	Description
1	La sortie du PID est forcée à la valeur Max du champ Limites dans l'onglet, page 176 Sortie jusqu'à ce que la valeur de processus atteigne Point de consigne + h.
2	La phase 2 de l'Auto-Tuning comprend deux étapes : 1. La sortie du PID est forcée à la valeur Min du champ Limites dans l'onglet, page 176 Sortie jusqu'à ce que la valeur de processus atteigne Point de consigne - h. 2. La sortie du PID est forcée à la valeur Max du champ Limites dans l'onglet, page 176 Sortie jusqu'à ce que la valeur de processus atteigne Point de consigne + h.
3	La sortie du PID est forcée à la valeur Min du champ Limites dans l'onglet, page 176 Sortie jusqu'à ce que la valeur de processus atteigne Point de consigne - h.
4	La phase 4 de l'Auto-Tuning comprend deux étapes : 1. La sortie du PID est forcée à la valeur Max du champ Limites dans l'onglet, page 176 Sortie jusqu'à ce que la valeur de processus atteigne Point de consigne + h. 2. La sortie du PID est forcée à la valeur Min du champ Limites dans l'onglet, page 176 Sortie , les paramètres du PID sont calculés et le PID devient actif.
(1) La dernière sortie appliquée au processus avant le début de l'Auto-Tuning est utilisée à la fois comme point de départ et comme point de relâche du processus Auto-Tuning.	

NOTE: Les paramètres K_p , T_i et T_d ne peuvent pas être calculés si le contrôle manuel des sorties est activé lors du processus d'étalonnage de l'Auto-Tuning. Lancez à nouveau le processus d'étalonnage de l'Auto-Tuning une fois que le contrôle manuel des sorties est terminé.

Stockage des coefficients calculés

Une fois la séquence d'Auto-Tuning terminée, les mots mémoire affectés aux coefficients K_p , T_i et T_d et au type d'action sont définis à l'aide des valeurs calculées. Ces valeurs sont consignées dans la mémoire RAM et enregistrées dans le contrôleur logique tant que l'application est correcte et qu'aucun démarrage à froid n'est effectué (%S0).

Si le système n'est pas influencé par des perturbations extérieures, les valeurs calculées peuvent être inscrites dans les paramètres du contrôleur *PID* (voir l'onglet **PID** de l'**Assistant PID**, page 175). Ainsi, le contrôleur *PID* peut être configuré en mode de marche *PID*.

Réglage des paramètres de PID

La méthode Auto-Tuning peut parfois fournir une commande très dynamique, conduisant à des dépassements indésirables pendant le changement des consignes. Pour affiner la régulation du processus fournie par les paramètres *PID* (K_p , T_i , T_d) provenant de l'Auto-Tuning, vous pouvez également ajuster ces valeurs manuellement, soit directement à partir de l'onglet **PID** de l'écran **Assistant PID**, soit via les mots mémoire (%MW) correspondants. Pour plus d'informations sur le réglage manuel des paramètres, consultez les annexes, page 183.

Lancement du processus Auto-Tuning

Dans l'onglet **AT**, le **Déclencheur AT** permet de répéter la séquence d'Auto-Tuning. Le processus d'Auto-Tuning est lancé à chaque front montant du signal lié au **Déclencheur AT**.

Pour configurer l'auto-tuning, reportez-vous à la rubrique Onglet **AT**, page 174.

Limites de l'utilisation d'Auto-Tuning

Les processus thermiques sont souvent assimilables au modèle de premier ordre avec temporisation pure. Deux paramètres principaux décrivent ce type de modèle :

- la constante de temps (τ),
- le délai de temporisation (θ).

L'**Auto-Tuning** est idéal pour les processus dans lesquels la constante de temps (τ) et le délai de temporisation (θ) répondent aux critères suivants :

- $10 \text{ s} < (\tau + \theta) < 2700 \text{ s}$ (soit : 45 minutes)
- $2 < \tau / \theta < 20$

Mode Manuel

Introduction

Le mode manuel est accessible dans l'écran **Assistant PID** (**Output** tab, page 175). Il vous permet de contourner les ordres du *PID*. Le mode manuel a deux objectifs principaux :

- initialiser la configuration,
- déterminer la période d'échantillonnage

Description

Le mode manuel vous permet de spécifier la **Output value**, page 175 Cette opération est idéale pour tester la réponse du système.

La définition de l'**Adresse de bit** sur 1 dans l'onglet **Sortie**, page 175 active le mode manuel. Si l'option **Activer** est définie, le mode manuel est le seul mode accessible.

Application

Lorsque le mode manuel est actif, la sortie prend la valeur fixe que vous configurez. Cette sortie est comprise entre 0 et 10 000 (0 à 100 % pour la sortie PWM).

Vous pouvez également utiliser le mode manuel pour déterminer la limite de sortie minimum/maximum par approximation.

Par ailleurs, le mode manuel est obligatoire pour utiliser la méthode de courbe de réponse du processus, page 166, qui permet de trouver la période d'échantillonnage (T_s) correcte.

Démarrage du mode manuel

Avant de passer en mode manuel, vérifiez que le commutateur RUN/STOP du Logic Controller est en position RUN.

Pour passer en mode manuel à l'aide d'une table d'animation, procédez comme suit :

Etape	Description
1	Activez le mode manuel en réglant le bit mémoire dédié sur 1. Pour plus d'informations, consultez l'onglet Sortie , page 175.
2	Si vous utilisez PWM, définissez la Période PWM sur la valeur souhaitée.
3	Régalez le mot mémoire associé au mode de fonctionnement dans l'onglet Général , page 169 de l' Assistant PID sur 1 (mode <i>PID</i>). Pour plus de détails sur les modes de fonctionnement avec le paramètre Adresse de mot, consultez la description du mode de fonctionnement, page 154.
4	Régalez le mot mémoire associé à la sortie manuelle dans l'onglet Sortie , page 175 sur la valeur souhaitée. Cette valeur de consigne peut être sélectionnée plusieurs fois, à condition que le système conserve son état initial.
5	Activez le contrôleur, page 158 de boucle.

Arrêt du mode manuel

Pour arrêter le mode manuel à l'aide d'une table d'animation, procédez comme suit :

Etape	Description
1	Désactivez le contrôleur, page 158 de boucle.
2	Inhibez le mode manuel en réglant le bit mémoire dédié sur 0. Pour plus d'informations, consultez l'onglet Sortie , page 175.
3	Réglez le mot mémoire associé au mode de fonctionnement sur 0 dans l'onglet Général , page 169 du contrôleur <i>PID</i> . Pour plus d'informations sur les modes de fonctionnement avec le paramètre Adresse de mot, consultez la description des modes de fonctionnement, page 154.
4	Réglez le mot mémoire associé à la sortie manuelle sur 0 dans l'onglet Sortie , page 175.

Détermination de la période d'échantillonnage (T_s)

Introduction

La période d'échantillonnage (T_s) est le paramètre principal de la régulation *PID*. La période d'échantillonnage (T_s) doit être définie avec soin dans l'onglet **PID**, page 172 de l'écran **Assistant PID**. Ce paramètre est étroitement corrélé à la constante de temps (τ) du processus à réguler.

Cette section décrit l'utilisation du mode en ligne et les deux méthodes permettant de déterminer la période d'échantillonnage (T_s) :

- Méthode de la courbe de réponse du processus
- Méthode des essais et erreurs

Méthode de la courbe de réponse du processus

Cette méthode en boucle ouverte permet de déterminer la constante de temps du processus à réguler. Tout d'abord, il convient de vérifier que le processus peut être décrit par un modèle de premier ordre avec délai de temporisation. Le principe est relativement simple. Il consiste à appliquer un changement d'étape à l'entrée du processus tout en enregistrant la courbe de sortie du processus. Ensuite, on utilise une méthode graphique pour déterminer le délai de temporisation du processus.

Pour déterminer la période d'échantillonnage (T_s) à l'aide de la méthode de la courbe de réponse du processus, procédez comme suit :

Etape	Action
1	Supposons que vous ayez déjà configuré les différents paramètres dans les onglets Général , Entrée , PID , AT et Sortie du <i>PID</i> .
2	Sélectionnez l'onglet Sortie , page 175 dans l'écran Assistant PID .
3	Sélectionnez Activer ou Adresse de bit dans la liste déroulante Mode manuel pour autoriser la sortie manuelle.
4	Réglez le champ Sortie sur un niveau élevé (compris entre 5000 et 10000).
5	Chargez votre application dans le Logic Controller. Pour plus d'informations sur le chargement d'une application, reportez-vous au document <i>EcoStruxure Machine Expert - Basic – Guide d'utilisation</i> .
6	Exécutez le <i>PID</i> et vérifiez l'augmentation de la courbe de réponse.
7	Lorsque la courbe de réponse a atteint un état stable, arrêtez la mesure du <i>PID</i> .
8	Utilisez la méthode graphique suivante pour déterminer la constante de temps (τ) du processus de contrôle : <ol style="list-style-type: none"> 1. Calculez la sortie de valeur de processus pour une augmentation de 63 % ($S_{[63\%]}$) à l'aide de la formule suivante : $S_{[63\%]} = S_{[initial]} + (S_{[final]} - S_{[initial]}) \times 63\%$ 2. Calculez graphiquement l'abscisse du temps ($t_{[63\%]}$) correspondant à $S_{[63\%]}$. 3. Calculez graphiquement le temps initial ($t_{[initial]}$) correspondant au début de l'augmentation de la réponse du processus. 4. Calculez la constante de temps (τ) du processus de régulation en utilisant la relation suivante : $\tau_{[63\%][initial]}$
9	Calculez la période d'échantillonnage (T_s) ⁽¹⁾ en fonction de la valeur de (τ) que vous avez déterminée à l'étape précédente, en utilisant la règle suivante : $T_s = \tau/75$
10	Définissez la Période de scrutation du Mode de scrutation périodique de telle sorte que la période d'échantillonnage (T_s) soit un multiple exact de la période de scrutation : Période de scrutation = T_s / n , où n est un entier positif ⁽²⁾
<p>(1) L'unité de base pour la période d'échantillonnage est 10 ms. Par conséquent, vous devez arrondir la valeur de T_s aux 10 ms les plus proches.</p> <p>(2) Vous devez choisir « n » pour que la Période de scrutation résultante soit un entier positif compris entre 1 et 150 ms.</p>	

Méthode des essais et erreurs

Cette méthode consiste à définir la période d'échantillonnage de la fonction Auto-Tuning par approximations successives, jusqu'à ce que l'algorithme converge vers des valeurs satisfaisantes de K_p , T_i et T_d .

NOTE: Contrairement à la méthode de courbe de réponse du processus, la méthode des essais et erreurs n'est basée sur aucune loi d'approximation de la réponse au processus. Cependant, elle présente l'avantage de converger vers une valeur de la période d'échantillonnage, du même ordre de grandeur que la valeur actuelle.

Pour définir la valeur d'Auto-Tuning par cette méthode, procédez comme suit :

Etape	Action
1	Sélectionnez l'onglet AT dans la fenêtre de configuration du <i>PID</i> .
2	Réglez la limitation de sortie de l'Auto-Tuning sur 10 000 .
3	Chargez votre application dans le Logic Controller. Pour plus d'informations sur le chargement d'une application, reportez-vous au document <i>EcoStruxure Machine Expert - Basic – Guide d'utilisation</i> .
4	Sélectionnez l'onglet PID dans l'écran Assistant PID .
5	Indiquez la première ou la n ^{ème} hypothèse dans le champ Période d'échantillonnage ⁽¹⁾ .
6	Lancez Auto-Tuning, page 155.
7	Patiencez jusqu'à la fin du processus Auto-Tuning.
8	Deux cas peuvent se produire : <ul style="list-style-type: none"> • Auto-Tuning se termine avec succès : Passez à l'étape 10. • Echec de l'Auto-Tuning : Reportez-vous à la section <i>Codes d'erreur détectée d'Auto-Tuning</i>, page 181. Cela signifie que l'approximation de la période d'échantillonnage (Ts) est incorrecte. Essayez une nouvelle valeur de Ts et répétez les étapes 3 à 8 autant de fois que nécessaire jusqu'à ce que le processus Auto-Tuning converge.
9	Suivez ces instructions pour fournir une nouvelle valeur de Ts : <ul style="list-style-type: none"> • Auto-Tuning génère le code d'erreur détectée 800C hex. Cela signifie que la période d'échantillonnage (Ts) est trop grande. Diminuez la valeur de Ts. • Auto-Tuning génère le code d'erreur détectée 800A hex. Cela signifie que la période d'échantillonnage (Ts) est trop petite. Augmentez la valeur de Ts.
10	Si nécessaire, ajustez les paramètres de contrôle <i>PID</i> ⁽²⁾ (Kp, Ti et Td) dans l'onglet PID , page 172 de l' Assistant PID .
<p>(1) Si vous n'avez pas de première indication sur la plage possible de la période d'échantillonnage, optez pour une valeur la plus petite possible : 1 (1 unité de 10 ms).</p> <p>(2) Si la régulation PID fournie par cet ensemble de paramètres de contrôle ne produit pas de résultats totalement satisfaisants, vous pouvez encore affiner l'évaluation de la période d'échantillonnage par essais et erreurs jusqu'à obtenir les paramètres de contrôle Kp, Ti et Td corrects.</p>	

Mode en ligne

En mode en ligne, lorsque le Logic Controller exécute la tâche périodique, la valeur affichée dans le champ Ts (de l'écran, page 168 **PID Assistant**) peut être différente du paramètre saisi (%MW). La valeur Ts est un multiple de la tâche périodique, tandis que la valeur %MW est celle qui est lue par le Logic Controller.

Assistant PID

Accès à l'Assistant PID

Introduction

Utilisez la fenêtre **Assistant PID** de EcoStruxure Machine Expert - Basic pour configurer le contrôleur *PID*


Assistant de configuration

Dans le tableau des propriétés *PID*, cliquez sur le bouton **Configuration [..]**. L'écran **Assistant PID** s'affiche.

Cette illustration montre l'écran **Assistant PID** :

L'écran **Assistant PID** affiche plusieurs onglets, selon que vous êtes en mode hors ligne ou en ligne :

Onglet	Mode d'accès	Liaison
Généralités	Hors ligne	Onglet Général, page 169
Entrée	Hors ligne	Onglet Entrée, page 171
PID	Hors ligne	Onglet PID, page 172
AT	Hors ligne	Onglet AT, page 174
Sortie	Hors ligne	Onglet Sortie, page 175

Dès qu'un mode de fonctionnement est sélectionné, les onglets contenant des champs vides à renseigner s'affichent comme suit  avec une bordure en rouge.

Onglet Général

Introduction

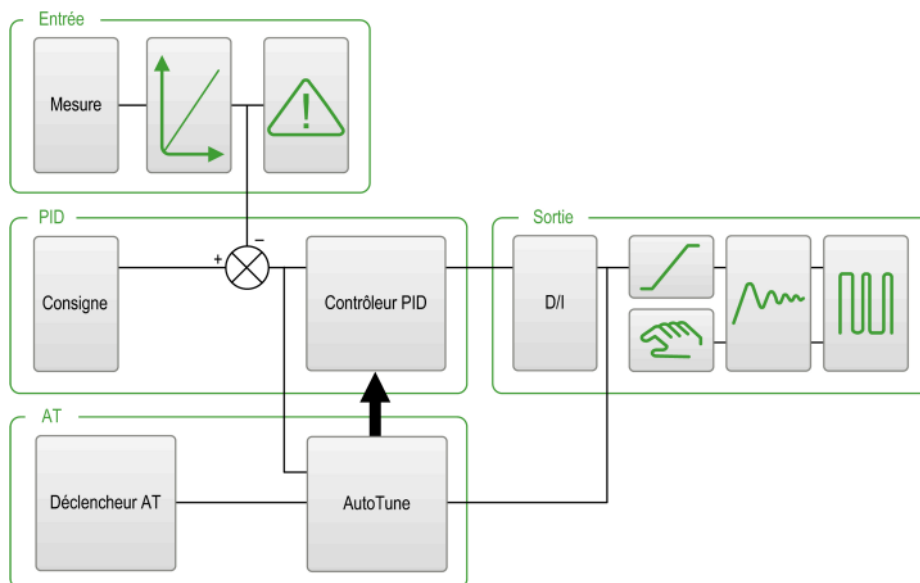
Cette section décrit l'onglet **Général** du *PID*. L'onglet **Général** s'affiche par défaut lorsque vous accédez à l'Assistant *PID* en mode hors ligne.

Description

Le tableau suivant décrit les paramètres de l'onglet **Général**.




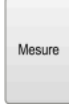

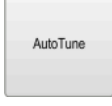

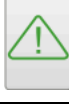




Paramètre	Description
Mode opératoire	Représente le mode <i>PID</i> à utiliser : <ul style="list-style-type: none"> • Non configuré • PID • AT + PID • AT • Adresse de mot Pour plus d'informations sur les modes de fonctionnement, consultez la section <i>Mode de fonctionnement PID</i> , page 154.
Adresse de mot	Dans ce champ (%MWxx), vous pouvez spécifier le mot mémoire utilisé pour configurer le mode de fonctionnement dans le programme. Le mot mémoire admet plusieurs valeurs en fonction du mode de fonctionnement que vous souhaitez définir : <ul style="list-style-type: none"> • %MWx = 0 (PID désactivé) • %MWx = 1 (pour configurer PID uniquement) • %MWx = 2 (pour configurer PID+Auto tuning) • %MWx = 3 (pour configurer Auto tuning uniquement) • %MWx = 4 (pour configurer PI uniquement)
Etats de PID	Si vous cochez cette case pour activer l'option, vous pouvez fournir un mot mémoire dans le champ associé (%MWxx) qui est utilisé par le contrôleur <i>PID</i> pour stocker l'état <i>PID</i> actuel pendant l'exécution du contrôleur <i>PID</i> et/ou de la fonction Auto-Tuning. Pour plus d'informations, consultez la section <i>Etats de PID et codes d'erreur détectée</i> , page 179.

Assistant graphique



L'assistant graphique permet de visualiser la façon dont la fonction *PID* est conçue. Ce graphique dynamique est mis à jour en fonction de la configuration.

Les icônes suivantes indiquent lorsqu'il est accessible ou ce qui se passe si vous cliquez dessus :

Ecran	Description
	Cliquez sur ce bouton pour afficher le champ Point de consigne de l'onglet PID , page 172.
	Cliquez sur ce bouton pour afficher l'onglet PID , page 172.
	Cliquez sur ce bouton pour afficher l'onglet Sortie , page 175.
	Cliquez sur ce bouton pour afficher l'onglet Entrée , page 171.
	Cliquez sur ce bouton pour afficher l'onglet AT , page 174.
	Cliquez sur ce bouton pour afficher l'onglet AT , page 174.
	Ce bouton s'affiche lorsque l'option Autoriser est cochée dans la zone Conversion de l'onglet Entrée , page 171.
	Ce bouton s'affiche lorsque l'option Autoriser est cochée dans la zone Alarmes de l'onglet Entrée , page 171.
	Ce bouton apparaît lorsque l'option Limites n'est pas configurée sur Désactiver dans la zone Limites de l'onglet Sortie , page 175.
	Ce bouton apparaît lorsque l'option Mode manuel n'est pas configurée sur Désactiver dans la zone Mode manuel de l'onglet Sortie , page 175.
	Cliquez sur ce bouton pour afficher l'onglet Sortie , page 175.
	Ce bouton apparaît lorsque l'option Autoriser est cochée dans la zone PWM sortie de l'onglet Sortie , page 175.

Onglet Entrée

Introduction

Cette section décrit l'onglet **Entrée** du *PID*. L'onglet **Entrée** permet de saisir les paramètres de l'entrée *PID*.

Cet onglet n'est accessible qu'en mode hors ligne et lorsqu'un mode de fonctionnement est sélectionné dans l'onglet **Général**.

Description

Le tableau suivant décrit les paramètres que vous pouvez définir.

Paramètre	Description	
Mesure	Indiquez la variable qui contient la valeur de processus à réguler. La plage par défaut est comprise entre 0 et 10000. Vous pouvez saisir un mot mémoire (%MWxx) ou une entrée analogique.	
Conversion	Autoriser	Cochez cette case pour convertir la valeur de processus [0...10000] en une plage linéaire Min.... Max.. La conversion s'applique également à la valeur de consigne.
	Min. Max.	Indiquez les valeurs minimum et maximum de l'échelle de conversion. La valeur de processus est ensuite automatiquement remise à l'échelle au sein de l'intervalle Min.... Max. Min. ou Max. peuvent être des mots mémoire (%MWxx), des mots constants (%KWxx) ou une valeur de -32768 à +32767. NOTE: Remarque : La valeur Min. doit être inférieure à la valeur Max..
Filtre	Autoriser	Cochez cette case pour appliquer un filtre à l'entrée mesurée.
	(100 ms)	Spécifiez la valeur du filtre (de 0 à 10000) ou une adresse de mot mémoire (%MWxx). L'unité de base de temps du filtre est 100 ms.
Alarmes	Autoriser	Cochez cette case pour activer les alarmes dans les variables d'entrée. Les valeurs d'alarme sont à déterminer par rapport à la valeur de processus obtenue après la phase de conversion. Les valeurs d'alarme doivent être comprises entre Min. et Max. lorsque la conversion est active. Sinon, elles seront comprises entre 0 et 10000.
	Faible Sortie	Indiquez la valeur d'alarme basse dans le champ Faible. Cette valeur peut être un mot mémoire (%MWxx), une constante (%KWxx) ou une valeur directe. Sortie doit contenir l'adresse du bit qui sera mis à 1 une fois la limite basse atteinte. Sortie peut être un bit mémoire (%Mxx) ou une sortie.
	Elevé Sortie	Indiquez la valeur d'alarme haute dans le champ Elevé. Cette valeur peut être un mot mémoire (%MWxx), une constante (%KWxx) ou une valeur directe. Sortie doit contenir l'adresse du bit qui sera mis à 1 une fois la limite haute atteinte. Sortie peut être un bit mémoire (%Mxx) ou une sortie.

Onglet PID

Introduction

Utilisez l'onglet **PID** pour entrer les paramètres de *PID* interne.

Cet onglet n'est accessible qu'en mode hors ligne et si un mode de fonctionnement est sélectionné dans l'onglet **Général**.

Description

Le tableau suivant décrit les paramètres que vous pouvez définir :

Paramètre	Description
Consigne	<p>Spécifiez la valeur de consigne du <i>PID</i>. Cette valeur peut être un mot mémoire (%<i>MWxx</i>), un mot constant (%<i>KWxx</i>) ou une valeur directe.</p> <p>Cette valeur doit donc être comprise entre 0 et 10000 lorsque la conversion est inhibée. Sinon, elle doit être comprise entre les valeurs Min. et Max. de la conversion.</p>
Type de correcteur	<p>Si le mode de fonctionnement PID ou AT + PID a été choisi au préalable dans le tableau des propriétés <i>PID</i>, vous pouvez sélectionner le type de correcteur souhaité (PID ou PI) dans la liste déroulante. Si d'autres modes de fonctionnement (AT ou Adresse de mot) ont été choisis, le champ Type de correcteur affiche Auto et devient grisé (non modifiable manuellement).</p> <p>Si vous sélectionnez PI dans la liste déroulante, le paramètre <i>Td</i> est forcé à 0 et ce champ est désactivé.</p>
Paramètres ⁽¹⁾	<p>Kp (x 0,01 s)</p> <p>Indiquez le gain proportionnel <i>PID</i>, multiplié par 100.</p> <p>Cette valeur peut être un mot mémoire (%<i>MWxx</i>), un mot constant (%<i>KWxx</i>) ou une valeur directe.</p> <p>La plage valide pour le paramètre <i>Kp</i> est : $0 < Kp < 10000$.</p> <p>NOTE: Si <i>Kp</i> est défini par erreur sur 0 ($Kp \leq 0$ est non valide), la valeur par défaut $Kp = 100$ est automatiquement affectée par la fonction <i>PID</i>.</p>
	<p>Ti (x 0,1 s)</p> <p>Spécifiez le temps intégral pour une base de temps de 0,1 seconde.</p> <p>Cette valeur peut être un mot mémoire (%<i>MWxx</i>), un mot constant (%<i>KWxx</i>) ou une valeur directe.</p> <p>Elle doit être comprise entre 0 et 36000.</p> <p>NOTE: Pour désactiver l'action intégrale de la fonction <i>PID</i>, réglez ce coefficient sur 0.</p>
	<p>Td (x 0,1 s)</p> <p>Spécifiez le temps dérivé pour une base de temps de 0,1 seconde.</p> <p>Cette valeur peut être un mot mémoire (%<i>MWxx</i>), un mot constant (%<i>KWxx</i>) ou une valeur directe.</p> <p>Elle doit être comprise entre 0 et 10000.</p> <p>NOTE: Pour désactiver l'action dérivée de la fonction <i>PID</i>, réglez ce coefficient sur 0.</p>
Période d'échantillonnage	<p>Indiquez ici la période d'échantillonnage de la fonction <i>PID</i> pour une base de temps de 10^{-2} secondes (10 ms).</p> <p>Cette valeur peut être un mot mémoire (%<i>MWxx</i>), un mot constant (%<i>KWxx</i>) ou une valeur directe.</p> <p>Elle doit être comprise entre 1 (0,01 s) et 10000 (100 s).</p>
<p>(1) Lorsque Auto-Tuning est activé, il n'est plus nécessaire de définir les paramètres <i>Kp</i>, <i>Ti</i> et <i>Td</i>, car ils le sont automatiquement par l'algorithme Auto-Tuning. Dans ce cas, vous ne devez saisir dans ces champs qu'une adresse de mot interne (%<i>MWxx</i>). N'entrez pas de mot constant ni de valeur directe lorsque Auto-Tuning est activé.</p>	

Onglet AT

Introduction

L'onglet **AT** concerne la fonction Auto-Tuning. Pour plus d'informations, consultez la section Réglage du PID avec Auto-Tuning, page 161.

Cet onglet n'est accessible qu'en mode hors ligne et si un mode de fonctionnement est sélectionné dans l'onglet **Général**.

Description

L'auto-tuning de la fonction PID est un processus en boucle ouverte qui agit directement sur le processus de contrôle sans régulation ni autre limitation que celles définies par la variable de processus (PV) et la consigne de sortie. Vous devez donc sélectionner avec soin les deux valeurs dans la plage autorisée par le processus, afin d'éviter toute surcharge éventuelle.

Lorsque le PID est mis en œuvre avec Auto-Tuning, le paramètre **Correcteur AT dynamique** affecte la valeur du gain proportionnel (Kp). Le calcul du gain proportionnel dans le processus Auto-Tuning dépend de la vitesse du correcteur dynamique sélectionné. Vous pouvez sélectionner l'une des options suivantes :

- **Rapide**
- **Moyenne**
- **Lente**
- **Adresse de mot**

Reportez-vous à la description de ces options dans le tableau ci-dessous.

▲ AVERTISSEMENT

FONCTIONNEMENT PID INSTABLE

- La limite de la valeur de processus (PV) et les valeurs de consigne de sortie doivent être définies en pleine connaissance de leurs effets sur la machine ou le processus.
- Pour la valeur de processus et la consigne de sortie, gardez des valeurs au sein de la plage admissible.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

▲ AVERTISSEMENT

FONCTIONNEMENT IMPRÉVU DE L'ÉQUIPEMENT

N'utilisez pas de sortie à relais avec la fonction PID.

Le non-respect de ces instructions peut provoquer la mort, des blessures graves ou des dommages matériels.

Le tableau suivant décrit les paramètres que vous pouvez définir :

Champ	Description	
Mode AT	Autoriser	<p>Cochez cette case pour activer l'opération Auto-Tuning.</p> <p>Vous pouvez utiliser cette case à cocher de deux façons, selon que vous définissez le mode de fonctionnement manuellement ou via une adresse de mot dans l'onglet Général de la fonction <i>PID</i> :</p> <ul style="list-style-type: none"> • Si vous réglez Mode de fonctionnement sur PID + AT ou AT dans l'onglet Général, page 169, l'option Autoriser est activée et n'est pas modifiable. • Si vous sélectionnez le mode de fonctionnement via une adresse de mot <i>%MWx</i> (<i>%MWx</i> = 2 : PID + AT ; <i>%MWx</i> = 3 : AT), vous devez cocher l'option Autoriser manuellement pour pouvoir configurer les paramètres d'Auto-Tuning.
Plage de mesure	Autoriser	<p>Cochez cette case pour activer la plage de mesure.</p> <p>NOTE: Si la plage de mesure est désactivée, la valeur Min. est réglée à 0 et la valeur Max. à 10 000.</p>
	Min. Max.	<p>Paramétrez les valeurs Min. et Max en fonction de la plage de mesures de 1 % au-dessus ou en dessous du point de consigne.</p> <p>Les valeurs peuvent être immédiates entre 1 et 10 000 ou un mot mémoire <i>%MWx</i>.</p> <p>NOTE: la valeur Min. doit être inférieure à la valeur Max.</p> <p>Exemple : Si la valeur de processus doit se situer autour de 35 °C ± 3 °C :</p> <ul style="list-style-type: none"> • Le point de consigne est 350. • ± 3 °C est h, page 163 et doit être 30. • Donc, 1 % × (Max. – Min.) = 30 • Soit 1 % × 3000 = 30 • D'où Max. = 3100 et Min. = 100
Correcteur AT dynamique	Rapide Moyenne Lent Adresse de mot	<p>Il affecte la valeur du gain proportionnel (Kp), calculée par le processus AT.</p> <ul style="list-style-type: none"> • Rapide offre un temps de réponse rapide avec davantage de dépassements que la moyenne. • Moyenne offre un temps de réponse moyen avec un dépassement moyen. • Lente offre un temps de réponse plus lent avec moins de dépassements que la moyenne. • Adresse de mot fournit le temps de réponse configuré dans l'objet de mot spécifié (<i>%MW</i>).
Déclencheur AT	Déclencheur AT	<p>Ce paramètre vous permet de lancer le processus AT, chaque fois qu'un front montant est détecté sur le bit dédié (bit mémoire ou bit d'entrée numérique).</p> <p>Le processus AT s'arrête lorsqu'un front descendant est détecté sur l'entrée Déclencheur AT. De plus, en mode PID + AT, la régulation PID s'arrête également.</p>

Coefficients calculés Kp, Ti et Td

Une fois le processus Auto-Tuning terminé, les coefficients *PID* Kp, Ti et Td calculés sont stockés dans leurs mots mémoire (*%MWx*) respectifs.

Onglet Sortie

Introduction

Cet onglet permet de renseigner les paramètres de sortie de la fonction *PID*.

Cet onglet n'est accessible qu'en mode hors ligne et si un mode de fonctionnement est sélectionné dans l'onglet **Général**.

Description

Le tableau suivant décrit les paramètres que vous pouvez définir :

Champ	Description
Action	<p>Indiquez ici le type d'action de la fonction <i>PID</i> sur le processus. Trois options sont disponibles : Inverse, Direct et Adresse de bit. Si une augmentation de la sortie entraîne une augmentation de la mesure de la valeur de processus, choisissez l'action inversée (Inverse) ; si cela entraîne une diminution de la valeur de processus, choisissez une action de <i>PID</i> directe (Direct).</p> <p>Si vous sélectionnez Adresse de bit⁽¹⁾, vous pouvez modifier le type d'action en changeant le bit associé, lequel est soit un bit mémoire (%Mxx), soit une adresse d'entrée (%Ix.y).</p> <p>Le bit mémoire est mis à 1 si l'action sélectionnée est de type Direct et à 0 si l'action sélectionnée est de type Inverse.</p>
Limites	<p>Indiquez si des limites doivent être appliquées à la sortie de <i>PID</i>. Il y a 3 options disponibles : Activer, Désactiver et Adresse de bit.</p> <p>Sélectionnez Activer pour mettre le Bit à 1 ou sélectionnez Désactiver pour mettre le Bit à 0.</p> <p>Sélectionnez Adresse de bit pour gérer les limites de bit en modifiant le bit associé, lequel est soit un bit mémoire (%Mxx), soit une adresse d'entrée (%Ix.y).</p> <p>Définissez les limites haute et basse de la sortie <i>PID</i>.</p> <p>Min. ou Max. peut être un mot mémoire (%MWxx), un mot constant (%KWxx) ou une valeur de 1 à 10000 (0,01 % à 100 % de la période PWM).</p> <p>NOTE: La valeur Min. doit être inférieure à la valeur Max..</p>
Mode manuel	<p>Spécifiez si la fonction <i>PID</i> doit passer en mode manuel. Il y a 3 options disponibles : Activer, Désactiver et Adresse de bit.</p> <p>Si vous sélectionnez Adresse de bit, vous pouvez passer en mode manuel (bit à 1) ou automatique (bit à 0) à l'aide du programme, en modifiant le bit associé (bit mémoire (%Mxx) ou entrée).</p> <p>La Sortie du mode manuel doit contenir la valeur que vous désirez affecter à la sortie analogique lorsque le <i>PID</i> est en mode manuel, page 165. Cette Sortie peut être un mot (%MWxx) ou une valeur directe au format [0 à 10 000].</p>
Sortie analogique	<p>Spécifiez la sortie <i>PID</i> à utiliser en mode Auto-Tuning (AT).</p> <p>Cette Sortie analogique⁽²⁾ peut être une adresse de mot mémoire ou de sortie analogique. Lors de l'utilisation de la fonction PWM de <i>PID</i>, seules les adresses de mot mémoire sont autorisées.</p>
PWM sortie	<p>Cochez cette case pour utiliser la fonction PWM de <i>PID</i>.</p> <p>Indiquez la période de modulation dans la zone de texte Période (x 0,1 s). Cette période, obligatoirement comprise entre 1 et 500, peut être un mot mémoire (%MWxx) ou un mot constant (%KWxx). La précision de PWM dépend de la période PWM et de la période de scrutation. La précision augmente lorsque le ratio de PWM (<i>PWM.R</i>) a le plus grand nombre de valeurs. Par exemple, avec une période de scrutation de 20 ms et une période PWM de 200 ms, <i>PWM.R</i> peut prendre les valeurs 0 %, 10 %, 20 %, 30 %, 40 %, 50 %, 60 %, 70 %, 80 %, 90 %, 100 %. Avec une période de scrutation de 50 ms et une période de PWM de 200 ms, <i>PWM.R</i> peut prendre les valeurs 0 %, 25 %, 50 %, 75 % et 100 % de la période <i>PWM.P</i>.</p> <p>Exemple : cas PWM.R = 75 %</p> <p> T_s = période de scrutation T_{on} = somme des T_s </p> <p>Dans le champ Sortie, définissez le bit de sortie PWM. Il peut s'agir d'un bit mémoire (%Mxx) ou d'une adresse de sortie. Pour plus d'informations sur la fonction PWM, consultez le chapitre Pulse Width Modulation (%PWM), page 48.</p>
<p>(1) Lorsque la fonction Auto-Tuning est activée, son algorithme détermine automatiquement le type d'action (direct ou inverse) approprié au processus de contrôle. Vous devez ensuite entrer un bit mémoire (%Mxx) dans le champ Adresse de bit associé.</p> <p>(2) Entrez une adresse mémoire (%MWxx) ou une adresse de sortie analogique (%QWx.y).</p>	

Programmation du PID

Utilisation de la fonction PID

Cette section fournit des descriptions et des conseils de programmation pour utiliser la fonction **PID**.

Description

Introduction

Un *PID* (proportionnel-intégral-dérivé) est un mécanisme de boucle de contrôle générique (contrôleur) largement utilisé dans les systèmes de régulation industriels. Le *PID* utilise un algorithme qui met en œuvre trois paramètres constants : les valeurs proportionnelle, intégrale et dérivée, appelées P, I et D respectivement.

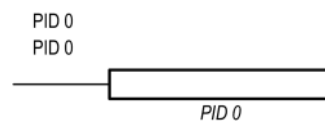
Caractéristiques principales

Les principales fonctionnalités de la fonction PID de EcoStruxure Machine Expert - Basic sont les suivantes :

- Entrée analogique
- Conversion linéaire de la mesure configurable
- Alarme haute et basse d'entrée configurable
- Sortie analogique ou PWM
- Coupure de la sortie configurable
- Action directe ou inverse configurable
- Fonction Auto-Tuning

Illustration

Voici la fonction *PID* dans l'éditeur LD (schéma à contacts) de EcoStruxure Machine Expert - Basic :



NOTE: Un espace doit séparer la mention PID de son numéro (par exemple, PID<espace>0).

Paramètres

Contrairement aux blocs fonction *Timer* ou *Counter*, il existe pas de bloc fonction *PID* dans EcoStruxure Machine Expert - Basic. L'instruction `[PID x]` n'active que la fonction de boucle de contrôle *PID*, x étant le numéro de PID.

Pour configurer la fonction *PID*, accédez à la fenêtre **Programmation**, cliquez sur **Outils > PID**, puis modifiez les propriétés du PID (reportez-vous au tableau ci-dessous pour les paramètres de configuration).

La fonction *PID* a les paramètres suivants :

Paramètre	Description	Valeur
Utilisée	Cochée si les E/S sont utilisées dans le projet.	True/False False (par défaut)
PID	Nom de l'objet <i>PID</i> en cours	Un programme ne peut contenir qu'un nombre limité de fonctions <i>PID</i> . Pour connaître le nombre maximum d'objets <i>PID</i> , consultez le tableau Nombre maximum d'objets (voir Modicon M221 Logic Controller - Guide de programmation).
Symbole	Symbole de l'objet <i>PID</i> en cours	Symbole associé à cet objet <i>PID</i> . Pour plus d'informations, reportez-vous à Définition et utilisation des symboles (voir EcoStruxure Machine Expert - Basic - Guide d'exploitation).
[...]	Bouton permettant de démarrer l'assistant	Cliquez dessus pour afficher l'écran Assistant PID . Pour plus d'informations, consultez la section Assistant PID, page 168.
Commentaire	Commentaire	Un commentaire peut être associé à cet objet.

Programmation et configuration

Introduction

Cette section décrit comment programmer et configurer le contrôleur EcoStruxure Machine Expert - Basic *PID*.

Activation du contrôleur PID

L'exemple suivant active la boucle du contrôleur *PID 0* si le bit *%M0* est défini sur 1 :

Réseau	Instruction
0	LD %M0 [PID 0]

NOTE: Pour obtenir le schéma à contacts équivalent, reportez-vous à la procédure de réversibilité (voir EcoStruxure Machine Expert - Basic, Guide de la bibliothèque des fonctions génériques).

Mesure analogique du PID

La fonction *PID* effectue une correction de *PID* à l'aide d'une mesure analogique et d'une consigne, puis elle produit une commande analogique au même format ou une PWM sur une sortie numérique.

Pour utiliser le *PID* à pleine mesure (résolution maximale), configurez l'entrée analogique dédiée à la mesure du contrôleur *PID* au format [0 à 10 000]. Toutefois, le contrôleur *PID* fonctionne correctement si vous utilisez la configuration par défaut [0 à 4095].

Configuration de la période de scrutation

Lorsque vous utilisez des contrôleurs *PID* EcoStruxure Machine Expert - Basic, vous devez configurer le mode de scrutation du contrôleur logique sur **Périodique**

(onglet **Programme, Tâches > Tâche maître**). En mode périodique, chaque scrutation du Logic Controller débute à intervalles réguliers, afin que le taux d'échantillonnage reste constant tout au long de la période de mesure. Pour plus d'informations sur la configuration du mode de scrutation, consultez le document *EcoStruxure Machine Expert - Basic - Guide d'utilisation*.

En mode périodique, le système met le bit système %S19 à 1 si la durée de scrutation du Logic Controller est supérieure à la période définie par le programme de l'utilisateur.

Etats de PID et codes d'erreur détectée

Introduction

Le contrôleur *PID* EcoStruxure Machine Expert - Basic permet d'écrire l'état actuel du contrôleur *PID* et du processus Auto-Tuning dans un mot mémoire défini par l'utilisateur. Pour plus d'informations sur l'activation et la configuration du mot mémoire des états de *PID*, consultez l'onglet **Général**, page 169 de l'Assistant PID, page 168.

Le mot mémoire des états de *PID* peut enregistrer les informations suivantes concernant le *PID* :

- Etat actuel du contrôleur *PID*
- Etat actuel du processus Auto-Tuning
- Codes d'erreur détectée du *PID*
- Codes d'erreur détectée d'Auto-Tuning

NOTE: Le mot mémoire des états est en lecture seule.

Mot mémoire des états

États	Description
0000 hex	Le contrôle <i>PID</i> est inactif.
2000 hex	Le contrôle <i>PID</i> est en cours.
4000 hex	La consigne <i>PID</i> est atteinte.

Mot mémoire des états d'Auto-Tuning

Etat d'Auto-Tuning	Description
0100 hex	La phase 1 de l'Auto-Tuning, page 163 est en cours.
0200 hex	La phase 2 de l'Auto-Tuning, page 163 est en cours.
0400 hex	La phase 3 de l'Auto-Tuning, page 163 est en cours.
0800 hex	La phase 4 de l'Auto-Tuning, page 163 est en cours.
1000 hex	Phase d'Auto-Tuning terminée

Codes d'erreur détectée de PID

Le tableau suivant décrit les erreurs susceptibles d'être détectées pendant le contrôle *PID* :

Code d'erreur détectée	Description
8001 hex	Valeur du mode de fonctionnement hors plage.
8002 hex	Les valeurs minimum et maximum de la conversion linéaire sont égales.
8003 hex	La limite supérieure de la sortie TOR est inférieure à la limite inférieure.
8004 hex	La limite de consigne se trouve hors de la plage de conversion linéaire.
8005 hex	La limite de consigne est inférieure à 0 ou supérieure à 10 000.
8006 hex	La consigne se trouve hors de la plage de conversion linéaire.
8007 hex	La consigne est inférieure à 0 ou supérieure à 10 000.
8008 hex	L'action de contrôle est différente de l'action déterminée au démarrage d'Auto-Tuning.

Codes d'erreur détectée d'Auto-Tuning

Le tableau suivant enregistre les messages d'erreur d'Auto-Tuning et décrit les causes possibles, ainsi que les actions de dépannage :

Code d'erreur détectée	Description
8009 hex	La limite de la valeur de processus est atteinte. Comme l'Auto-Tuning est un processus en boucle ouverte, la limite de la valeur de processus (PV) est la valeur maximale autorisée.
800A hex	La période d'échantillonnage est trop courte ou la consigne de sortie est trop basse. Augmentez la période d'échantillonnage ou la valeur de consigne de sortie d'Auto-Tuning.
800B hex	K_p est égal à 0.
800C hex	La constante de temps est négative. Il est possible que la période d'échantillonnage soit trop longue. Pour plus d'informations, consultez la section Limites de l'utilisation d'Auto-Tuning, page 184.
800D hex	Le délai est négatif.
800E hex	<p>Erreur détectée lors du calcul de K_p. L'algorithme Auto-Tuning est instable (pas de convergence). Causes possibles :</p> <ul style="list-style-type: none"> Des perturbations du processus pendant l'Auto-Tuning ont causé une distorsion du calcul du gain statique du processus. La réponse transitoire de la valeur de processus est insuffisante pour que l'Auto-Tuning puisse calculer le gain statique. Les deux. <p>Vérifiez les paramètres de <i>PID</i> et Auto-Tuning et effectuez des ajustements pour améliorer la convergence. Vérifiez également qu'aucune perturbation n'affecte la valeur de processus. Essayez de modifier :</p> <ul style="list-style-type: none"> la consigne de sortie, la période d'échantillonnage. <p>Assurez-vous que le processus n'est pas perturbé lorsque l'Auto-Tuning est en cours.</p>
800F hex	La constante de temps dépasse le ratio de délai ($\tau/\theta > 20$). La régulation du <i>PID</i> n'est plus stable. Pour plus d'informations, consultez la section Limites de l'utilisation d'Auto-Tuning, page 184.
8010 hex	La constante de temps dépasse le ratio de délai ($\tau/\theta < 2$). La régulation du <i>PID</i> n'est plus stable. Pour plus d'informations, consultez la section Limites de l'utilisation d'Auto-Tuning, page 184.
8011 hex	La limite du gain statique K_p a été dépassée ($K_p > 10000$). La sensibilité de la mesure de certaines variables d'application est peut-être trop faible. La plage doit être réévaluée dans l'intervalle [0 à 10000].
8012 hex	La valeur calculée de la constante de temps intégral (T_i) a été dépassée ($T_i > 20000$).
8013 hex	La valeur calculée de la constante de temps dérivé (T_d) a été dépassée ($T_d > 10000$).
8014 hex	Valeur de variable d'entrée non valide (hors de la plage définie par les alarmes, page 172 de sortie basse et sortie élevée).
8015 hex	<p>Erreur de traitement de filtre :</p> <ul style="list-style-type: none"> Durée de cycle hors plage. Temps de filtre $< 10 \times$ durée de cycle.

Annexes

Contenu de cette partie

Paramètres du PID	183
-------------------------	-----

Vue d'ensemble

Cette annexe reprend des extraits du guide de programmation aux fins de faciliter la compréhension technique de la documentation de la bibliothèque.

Paramètres du PID

Contenu de ce chapitre

Rôle et influence des paramètres PID	183
Méthode de réglage des paramètres PID	184

Rôle et influence des paramètres PID

Introduction

Cette section décrit le rôle et l'influence des paramètres PID.

Modèle de contrôleur PID

Le contrôleur PID EcoStruxure Machine Expert - Basic applique une correction PID mixte (série-parallèle). Les actions intégrale et dérivée s'exécutent indépendamment et en parallèle. L'action proportionnelle agit sur la sortie combinée des actions intégrale et dérivée.

Algorithmes de calcul

Deux algorithmes de calcul sont utilisés, selon la valeur de la constante de temps intégral (T_i) :

- Si $T_i \neq 0$, l'algorithme incrémentiel est utilisé.
- Si $T_i = 0$, un algorithme positionnel est utilisé et un décalage de +5000 est appliqué à la sortie PID.

Influence des actions

L'action proportionnelle sert à influencer la vitesse de réponse du processus. Une augmentation de l'action proportionnelle se traduit par :

- une réponse plus rapide,
- une erreur statique plus faible,
- une stabilité moindre.

L'action intégrale permet d'annuler l'erreur statique. Une augmentation de l'action d'intégration (c'est-à-dire une diminution du temps intégral T_i) se traduit par :

- une réponse plus rapide,
- une stabilité moindre.

L'action dérivée est anticipative. En pratique, elle ajoute un terme qui prend en compte la vitesse de variation de l'écart (ce qui permet d'anticiper les modifications en accélérant les temps de réponse du processus lorsque l'écart augmente et en les ralentissant lorsque l'écart diminue). Une augmentation de l'action dérivée (c'est-à-dire du temps dérivé) se traduit par :

- une réponse plus lente,
- un dépassement réduit.

NOTE: Compte tenu du temps dérivé, T_d est le temps utilisé pour anticiper la variation de l'écart. Les valeurs de T_d trop basses ou trop élevées peuvent provoquer des oscillations indésirables.

Pour chaque action, un compromis approprié doit être trouvé entre vitesse et stabilité.

Limites de la boucle de contrôle du PID

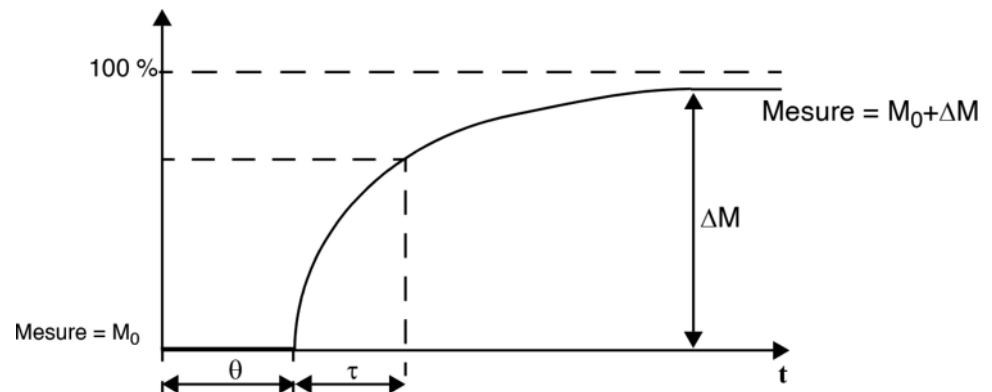
Le processus est assimilé à un premier ordre de temporisation pure avec une fonction de transfert :

$$H(p) = K \times \frac{e^{-\theta p}}{1 + \tau p}$$

où :

τ : constante de temps du modèle

θ : temporisation du modèle



Les performances de contrôle du processus dépendent du rapport $\frac{\tau}{\theta}$.

Le contrôle de processus PID approprié est atteint dans le domaine suivant : $2 < \frac{\tau}{\theta} < 20$.

Le contrôle PID convient particulièrement à la régulation des processus qui satisfont à la condition suivante :

- Pour $\frac{\tau}{\theta} < 2$, c'est-à-dire pour les boucles de contrôle rapides (faible θ) ou pour les processus à temporisation importante (τ élevé), le contrôle de processus PID n'est plus adapté. Dans ce cas, des algorithmes plus complexes doivent être utilisés.
- Pour $\frac{\tau}{\theta} > 20$, un contrôle de processus utilisant un seuil plus hystérésis est suffisant.

Méthode de réglage des paramètres PID

Introduction

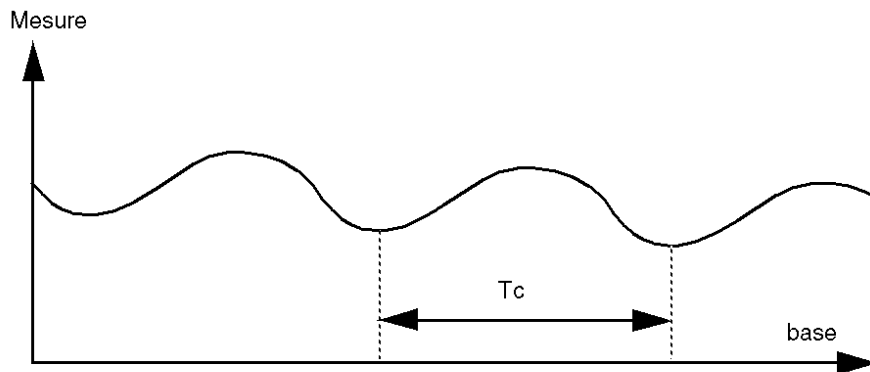
Plusieurs méthodes permettent de régler les paramètres PID. La méthode privilégiée est celle dite de Ziegler et Nichols, qui existe en deux variantes :

- réglage en boucle fermée ;
- réglage en boucle ouverte.

Avant d'appliquer l'une de ces méthodes, vous devez définir l'action, page 175 PID :

Réglage en boucle fermée

Ce principe utilise une commande proportionnelle ($T_i = 0, T_d = 0$) pour lancer le processus en augmentant un coefficient proportionnel jusqu'à ce qu'il recommence à osciller après l'application d'un niveau à la consigne du correcteur PID. Il suffit d'augmenter le gain proportionnel critique (K_{pc}) à l'origine de l'oscillation non amortie, et d'augmenter la période d'oscillation (T_c) pour réduire les valeurs et ainsi optimiser la régulation.

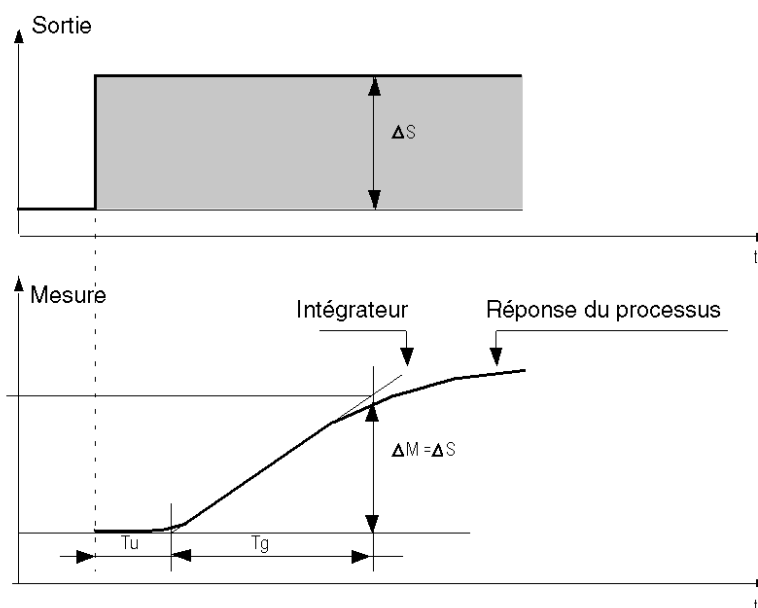


Selon le type de correcteur utilisé (PID ou PI), le réglage des coefficients est effectué avec les valeurs suivantes :

Correcteur	K_p : Gain proportionnel	T_i : Temps d'intégration	T_d : Dérivée
PID	$K_{pc}/1,7$	$T_c/2$	$T_c/8$
PI	$K_{pc}/2,22$	$0,83 \times T_c$	—

Réglage en boucle ouverte

Le régulateur étant en mode manuel, page 165, vous appliquez un niveau à la sortie et vous lancez la procédure de réponse comme pour un intégrateur avec un temps de retard pur.



Le point d'intersection à droite, représentant l'intégrateur avec les axes de temps, détermine le temps T_u . Le temps T_g est ensuite défini comme le temps

nécessaire pour que la variable contrôlée (mesure) ait la même taille de variation (% de l'échelle) que la sortie du régulateur.

Selon le type de correcteur utilisé (PID ou PI), le réglage des coefficients est effectué avec les valeurs suivantes :

Correcteur	Kp : Gain proportionnel	Ti : Temps d'intégration	Td : Dérivée
PID	-1,2 Tg/Tu	2 x Tu	0,5 x Tu
PI	-0,9 Tg/Tu	3,3 x Tu	–

NOTE: Pour plus d'informations sur les unités des paramètres, consultez l'onglet, page 172 **PID**.

Cette méthode de réglage fournit également une commande particulièrement dynamique qui peut s'exprimer par des dépassements indésirables lors du changement des impulsions des consignes. Dans ce cas, abaissez le gain proportionnel jusqu'à obtenir le comportement requis. Cette méthode ne requiert aucune hypothèse sur la nature et l'ordre de la procédure. Vous pouvez l'appliquer aussi bien aux procédures stables qu'aux procédures d'intégration réelles. Dans le cas de procédures lentes (comme dans l'industrie du verre), l'utilisateur n'a besoin que du début de la réponse pour réguler les coefficients Kp, Ti et Td.

A

accélération/décélération:

L'accélération est le taux de variation de la vitesse entre la **Vitesse de départ** et la vitesse cible. La décélération est le taux de variation de la vitesse entre la vitesse cible et la **Vitesse d'arrêt**. Ces variations de la vitesse sont implicitement gérées par la fonction PTO en fonction des paramètres d'accélération, de décélération et de jerk ratio, et selon un profil trapézoïdal ou de courbe en S.

C

CW/CCW:

ClockWise / Counter ClockWise

D

DWORD:

Abréviation de *double word*, mot double. Codé au format 32 bits.

F

fonction:

Unité de programmation possédant 1 entrée et renvoyant 1 résultat immédiat. Contrairement aux blocs fonction (FBs), une fonction est appelée directement par son nom (et non via une instance), elle n'a pas d'état persistant d'un appel au suivant et elle peut être utilisée comme opérande dans d'autres expressions de programmation.

Exemples : opérateurs booléens (AND), calculs, conversion (BYTE_TO_INT).

J

jerk ratio:

Proportion de modification de l'accélération et de la décélération, définie comme une fonction de temps.

M

mouvement absolu:

Mouvement vers une position définie par rapport à un point de référence.

P

POU:

Acronyme de *program organization unit*, unité organisationnelle de programme. Déclaration de variables dans le code source et jeu d'instructions correspondant. Les POU facilitent la réutilisation modulaire de programmes logiciels, de fonctions et de blocs fonction. Une fois déclarées, les POU sont réutilisables.

R

rampe à courbe en S:

Rampe d'accélération / de décélération où le paramètre `JerkRatio` est supérieur à 0 %.

rampe trapézoïdale:

Rampe d'accélération / décélération avec le paramètre `JerkRatio` défini sur 0%.

référencement:

Méthode utilisée pour déterminer le point de référence du mouvement absolu.

V**vitesse d'arrêt:**

Fréquence maximale à laquelle un moteur pas à pas arrête un mouvement, avec une charge appliquée et sans aucune perte de pas.

vitesse de démarrage:

Fréquence minimale à laquelle un moteur pas à pas peut produire un mouvement, avec une charge appliquée et sans aucune perte de pas.

Index

A

acquiescement et réinitialisation d'erreur	
MC_Reset_ATV	72
activation/désactivation de l'étage de puissance	
MC_Power_ATV	58
arrêt de mouvement	
MC_Stop_ATV	66

B

blocs fonction	
FC (Fast Counter)	23
générateur de fréquence (%FREQGEN)	149
HSC (High Speed Counter)	28
impulsion	41
MC_Halt_PTO	132
MC_Home_PTO	126
MC_Jog_ATV	60
MC_Motion_PTO	111
MC_MoveAbs_PTO	123
MC_MoveRel_PTO	120
MC_MoveVel_ATV	63
MC_MoveVel_PTO	117
MC_Power_ATV	58
MC_Power_PTO	114
MC_ReadMotionState_ATV	70
MC_ReadStatus_ATV	68
MC_Reset_ATV	72
MC_SetPost_PTO	128
MC_Stop_ATV	66
MC_Stop_PTO	130
modulation de largeur d'impulsion	48
MV_AbortTrigger_PTO	145
MV_ReadActPos_PTO	136
MV_ReadActVel_PTO	134
MV_ReadAxis_PTO	140
MV_ReadMotionState_PTO	139
MV_ReadPar_PTO	146
MV_ReadSts_PTO	137
MV_Reset_PTO	142
MV_TouchProbe_PTO	143
MV_WritePar_PTO	147
blocs fonction Drive	
configuration	58
blocs fonction Drive : codes d'erreur	75
BUFFER_MODE	104

C

codes d'erreur	
blocs fonction Drive	75
Codes d'objet de bloc fonction	
BUFFER_MODE	104
DIRECTION	103
HOMING_MODE	104
PTO_PARAMETER	105
compteur HSC	
mode fréquencemètre	38
compteur rapide	
configuration	24
description	23
exemple de programmation	26
compteur rapide HSC	
description	28

mode de comptage	32
configuring	
Drive function blocks	58

D

démarrage du mode jog, MC_Jog_ATV	60
déplacement à une vitesse spécifiée	
MC_MoveVel_ATV	63
DIRECTION	103
%DRV	53

F

%FC	23
fonctionnalités	
PTO	78
FREQGEN	
bloc fonction	149
configuration de bloc fonction	151

G

générateur de fréquence	
bloc fonction	149
gestion des entrées de bloc fonction et des objets d'entrée	
Execute	21
gestion des erreurs	
ErrID	21
Error	21
gestion des sorties de bloc fonction et des objets de sortie	
Busy	21
CmdAborted	21
Done	21
ErrID	21
Error	21

H

HOMING_MODE	104
%HSC	
compteur HSC	28

I

impulsion	
configuration de bloc fonction	42
description	41
exemple de programmation	47

J

JerkRatio	81
jeu	85

L

lecture de l'état de l'équipement	
%MC_ReadStatus_ATV	68
lecture de l'état du mouvement	
MC_ReadMotionState_ATV	70

M

MC_Halt_PTO	
arrêt contrôlé du mouvement jusqu'à la vitesse	
0	132
MC_Home_PTO	
commander à l'axe une séquence de	
référencement.....	126
MC_Jog_ATV	
démarrage du mode jog.....	60
MC_Motion_PTO	
appel d'une table de tâches de mouvement.....	111
MC_MoveAbs_PTO	
déplacement d'un axe vers une position donnée à	
une vitesse spécifiée	123
MC_MoveRel_PTO	
déplacement d'un axe sur une distance	
incrémentale à une vitesse spécifiée.....	120
MC_MoveVel_ATV	
déplacement à une vitesse spécifiée	63
MC_MoveVel_PTO	
déplacement d'un axe à une vitesse spécifiée	117
MC_Power_ATV	
activation/désactivation de l'étage de puissance...	58
MC_Power_PTO	
activation de l'alimentation d'un axe.....	114
MC_ReadMotionState_ATV	
lecture de l'état du mouvement.....	70
MC_ReadStatus_ATV	
lecture de l'état de l'équipement	68
MC_Reset_ATV	
acquittement et réinitialisation d'erreur.....	72
MC_SetPost_PTO	
déplacement de l'axe vers une position	
spécifiée	128
MC_Stop_ATV	
arrêt de mouvement	66
MC_Stop_PTO	
commande d'arrêt de mouvement contrôlé.....	130
modulation de largeur d'impulsion	
configuration de bloc fonction.....	49
description.....	48
exemple de programmation	52
motion task table	
PTO.....	88
MV_AbortTrigger_PTO	
abandon de blocs fonction connectés à des	
événements déclencheurs	145
MV_ReadActPos_PTO	
obtention de la position de l'axe.....	136
MV_ReadActVel_PTO	
obtention de la vitesse de l'axe.....	134
MV_ReadAxisError_PTO	
obtention d'une erreur de contrôle d'axe	140
MV_ReadMotionState_PTO	
obtention de l'état de mouvement de l'axe.....	139
MV_ReadPar_PTO	
obtention de paramètres à partir de la fonction	
PTO	146
MV_ReadSts_PTO	
obtention de l'état de l'axe.....	137
MV_Reset_PTO	
réinitialisation des erreurs liées à l'axe	142
MV_TouchProbe_PTO	
activation d'un événement déclencheur sur une	
entrée PROBE	143
MV_WritePar_PTO	
écriture de paramètres dans la fonction PTO.....	147

P

PID	
assistant de configuration	168
auto-tuning	155
configuration standard	158
description.....	177
états et codes d'erreur détectée	179
modes de fonctionnement.....	154
Onglet AT	174
Onglet Entrée	171
onglet Général	169
Onglet PID	172
onglet Sortie	175
paramètre.....	183
programmation et configuration.....	178
réglage en boucle fermée	185
réglage en boucle ouverte	185
%PLS	41
PTO	
configuration.....	88
fonctionnalités	78
motion task table.....	88
PTO_ERROR	106–107
PTO_PARAMETER.....	105
%PWM	48

R

rampe d'accélération	81
rampe de décélération	81

Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison
France

+ 33 (0) 1 41 29 70 00

www.se.com

Les normes, spécifications et conceptions pouvant changer de temps à autre, veuillez demander la confirmation des informations figurant dans cette publication.

© 2025 Schneider Electric. Tous droits réservés.

EIO0000003306.03