

Avant-propos

Introduction au S7-1200
puissant et souple

1

STEP 7 facilite le travail

2

Mise en route

3

Simplification des concepts
API

4

Création facile de la
configuration des appareils

5

Programmation facilitée

6

Communication aisée entre
les appareils

7

Le PID, c'est simple

8

Serveur Web pour une
connectivité Internet facile

9

La commande de
mouvement est simple

10

Utilisation aisée des outils en
ligne

11

IO-Link est facile

12

Caractéristiques techniques

A

Remplacement d'une CPU
V3.0 par une CPU V4.1

B

Mentions légales

Signalétique d'avertissement

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

DANGER

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

ATTENTION

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

PRUDENCE

signifie que la non-application des mesures de sécurité appropriées peut entraîner des blessures légères.

IMPORTANT

signifie que la non-application des mesures de sécurité appropriées peut entraîner un dommage matériel.

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

Personnes qualifiées

L'appareil/le système décrit dans cette documentation ne doit être manipulé que par du **personnel qualifié** pour chaque tâche spécifique. La documentation relative à cette tâche doit être observée, en particulier les consignes de sécurité et avertissements. Les personnes qualifiées sont, en raison de leur formation et de leur expérience, en mesure de reconnaître les risques liés au maniement de ce produit / système et de les éviter.

Utilisation des produits Siemens conforme à leur destination

Tenez compte des points suivants:

ATTENTION

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

Marques de fabrique

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

Exclusion de responsabilité

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

Avant-propos

Bienvenue dans l'univers du S7-1200. L'automate compact SIMATIC S7-1200 est le régulateur modulaire à faible encombrement, conçu pour les petits systèmes d'automatisation nécessitant une fonctionnalité simple ou avancée en ce qui concerne la logique, IHM et la mise en réseau. Sa forme compacte, son faible prix et ses caractéristiques performantes font du S7-1200 une solution idéale pour la commande de petites applications.

Dans le cadre de l'engagement SIMATIC pour une "automatisation totalement intégrée" (TIA), la gamme du S7-1200 et le logiciel de programmation TIA Portal vous offrent la flexibilité nécessaire pour répondre à vos besoins en automatisation.

Le S7-1200 contribue à faciliter les tâches les plus exigentes !

La solution de l'automate SIMATIC S7-1200, faisant partie de la catégorie de contrôleurs "compacts", est composée du contrôleur SIMATIC S7-1200 et des pupitres SIMATIC HMI Basic panels qui peuvent être programmés avec le logiciel d'ingénierie TIA Portal. La possibilité de programmer les deux appareils avec le même logiciel d'ingénierie réduit de manière significative les coûts de développement. Le portail TIA comprend STEP 7 pour la programmation du S7-1200 et WinCC pour la conception des projets de pupitre Basic panel.



Le contrôleur S7-1200 compact comporte :

- PROFINET intégré
- Des E/S rapides réalisant la commande de mouvement, des entrées analogiques intégrées permettant de réduire l'encombrement et la nécessité d'E/S supplémentaires, 4 générateurs d'impulsions pour les applications de train d'impulsions et de largeur d'impulsion (Page 73) et jusqu'à 6 compteurs rapides (Page 135)
- Des E/S intégrées aux modules unité centrale fournissant de 6 à 14 entrées et de 4 à 10 sorties



Des modules d'entrées-sorties pour tension continue, un relais ou des E/S analogiques augmentent le nombre d'E/S et des Signal Board innovants s'encliquettent sur la face avant de la CPU pour mettre à disposition des E/S supplémentaires (Page 18).

Les pupitres SIMATIC HMI Basic Panels (Page 20) ont été spécialement conçus pour le S7-1200.

Easy Book fournit une introduction à l'automate S7-1200. Les pages suivantes donnent une vue d'ensemble des nombreuses caractéristiques et fonctionnalités des appareils.

Pour plus d'informations, référez-vous au *Manuel système de l'automate programmable S7-1200*. Pour toute information concernant les certifications UL et FM, le marquage CE, C-Tick et d'autres normes, référez-vous aux Caractéristiques techniques (Page 381).

Ce manuel décrit les produits suivants :

- STEP 7 V13 SP1 Basic et Professional
- CPU S7-1200, version de firmware V4.1

Documentation et information

Le S7-1200 et STEP 7 proposent une grande variété de documents et autres ressources contenant les informations techniques dont vous avez besoin.

- Le manuel système de l'automate programmable S7-1200 fournit des informations spécifiques sur le fonctionnement, la programmation et les caractéristiques de toute la gamme du S7-1200. En plus du manuel système, le livret Easy Book du S7-1200 donne une vue d'ensemble plus générale des possibilités offertes par la gamme S7-1200.

Le manuel système et l'Easy Book sont disponibles au format électronique (PDF). Les manuels électroniques peuvent être téléchargés depuis le site Web du service client et se trouvent également sur le disque de documentation qui accompagne chaque CPU S7-1200.

- Le système d'information en ligne de STEP 7 offre un accès immédiat aux informations conceptuelles et aux instructions spécifiques qui décrivent le fonctionnement et les fonctionnalités du progiciel de programmation et le principe de fonctionnement des CPU SIMATIC.
- My Documentation Manager permet d'accéder aux versions électroniques (PDF) de l'ensemble de la documentation SIMATIC, y compris le manuel système, l'Easy Book et le système d'informations STEP 7. Avec My Documentation Manager, vous pouvez extraire des textes dans divers documents pour créer votre propre manuel personnalisé.

Le portail d'entrée du service client (<http://support.automation.siemens.com>) affiche un lien vers My Documentation Manager dans mySupport.

- Le site Web du service client fournit également des podcasts, FAQ et autres documents utiles concernant le S7-1200 et STEP 7. Les podcasts utilisent de courtes vidéos éducatives qui se concentrent sur des caractéristiques ou scénarios spécifiques afin de montrer les interactions, la commodité et l'efficacité offertes par STEP 7. Visitez les sites Web suivants pour accéder à la collection de podcasts :
 - Page Web du STEP 7 Basic (<http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/step7-basic/Pages/Default.aspx>)
 - Page Web du STEP 7 Professional (<http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/step7-professional/Pages/Default.aspx>)

- Vous pouvez également suivre ou rejoindre des discussions sur les produits sur le forum technique Service & Support (<https://www.automation.siemens.com/WW/forum/guests/Conferences.aspx?Language=en&siteid=csius&treeLang=en&group=4000002&extranet=standard&viewreg=WW&nodeid=34612486>). Ces forums vous permettent de dialoguer avec différents experts produits.
 - Forum pour le S7-1200 (<https://www.automation.siemens.com/WW/forum/guests/Conference.aspx?SortField=LastPostDate&SortOrder=Descending&ForumID=258&Language=en&onlyInternet=False>)
 - Forum pour STEP 7 Basic (<https://www.automation.siemens.com/WW/forum/guests/Conference.aspx?SortField=LastPostDate&SortOrder=Descending&ForumID=265&Language=en&onlyInternet=False>)

Service et assistance

En complément de notre documentation, Siemens propose son expertise technique sur Internet et via le site Web du service client (<http://www.siemens.com/tiaprotoal>).

Adressez-vous à votre agence Siemens si certaines de vos questions techniques restent sans réponse, si vous voulez connaître les offres de formation ou si vous désirez commander des produits S7. Comme ce personnel est techniquement formé et a des connaissances très pointues sur vos activités, vos processus et vos industries, ainsi que sur les différents produits Siemens que vous utilisez, il peut apporter les réponses les plus rapides et les plus efficaces possibles à tout problème que vous pourriez rencontrer.

Notes relatives à la sécurité

Siemens commercialise des produits et solutions comprenant des fonctions de sécurité industrielle qui contribuent à une exploitation sûre des installations, solutions, machines, équipements et/ou réseaux. Ces fonctions jouent un rôle important dans un système global de sécurité industrielle. Dans cette optique, les produits et solutions Siemens font l'objet de développements continus. Siemens vous recommande donc vivement de vous tenir régulièrement informé des mises à jour des produits.

Pour garantir une exploitation fiable des produits et solutions Siemens, il est nécessaire de prendre des mesures de protection adéquates (par ex. concept de protection des cellules) et d'intégrer chaque composant dans un système de sécurité industrielle global et moderne. Veuillez également tenir compte des produits que vous utilisez et qui proviennent d'autres fabricants. Pour plus d'informations sur la sécurité industrielle, rendez-vous sur (<http://www.siemens.com/industrialsecurity>).

Veuillez vous abonner à la newsletter d'un produit particulier afin d'être informé des mises à jour dès qu'elles surviennent. Pour plus d'informations, rendez-vous sur (<http://support.automation.siemens.com>).

Sommaire

Avant-propos	3
1 Introduction au S7-1200 puissant et souple	15
1.1 Introduction à l'automate S7-1200	15
1.2 Possibilités d'extension de la CPU	18
1.3 Modules S7-1200	19
1.4 Tableaux IHM basiques	20
1.5 Dimensions de montage et dégagement requis	21
1.6 Nouvelles fonctions	26
2 STEP 7 facilite le travail	29
2.1 Insertion facile d'opérations dans votre programme utilisateur	30
2.2 Accès facile à vos opérations favorites à partir d'une barre d'outils	31
2.3 Il est facile d'ajouter des entrées ou sorties aux opérations CONT et LOG	31
2.4 Instructions extensibles	32
2.5 Changement facile de l'état de fonctionnement de la CPU	33
2.6 Il est facile de modifier l'apparence et la configuration de STEP 7	33
2.7 Accès aisément aux bibliothèques du projet et bibliothèques globales	34
2.8 Sélection aisée de la version d'une instruction	34
2.9 Glisser-déplacer facilement entre les éditeurs	35
2.10 Modification du type d'appel pour un DB	36
2.11 Déconnexion temporaire d'appareils d'un réseau	37
2.12 "Retrait" virtuel facile des modules sans perte de configuration	38
3 Mise en route	39
3.1 Créer un projet	39
3.2 Créer des variables pour les E/S de la CPU	40
3.3 Créer un réseau simple dans votre programme utilisateur	42
3.4 Utiliser les variables API de la table des variables pour l'adressage des instructions	44
3.5 Ajouter une "boîte" d'opérations	45
3.6 Utilisez l'opération CALCULER pour une équation mathématique complexe	46
3.7 Ajouter un appareil IHM au projet	48
3.8 Créer une liaison réseau entre la CPU et l'appareil IHM	49
3.9 Créer une liaison IHM pour partager des variables	49

3.10	Créer un écran IHM	50
3.11	Sélectionner une variable API pour l'élément IHM	51
4	Simplification des concepts API	53
4.1	Tâches réalisées à chaque cycle.....	53
4.2	Etats de fonctionnement de la CPU.....	55
4.3	Exécution du programme utilisateur	57
4.3.1	Traitement du cycle à l'état MARCHE	57
4.3.2	Les OB vous aident à structurer votre programme utilisateur	58
4.3.3	Priorités d'exécution et mise en file d'attente des événements	58
4.4	Zones de mémoire, adressage et types de données.....	62
4.4.1	Types de données pris en charge par le S7-1200	63
4.4.2	Adressage des zones de mémoire	66
4.4.3	Accès à une "tranche" d'un type de données de variable	69
4.4.4	Accès à une variable par un type de données ajouté AT	71
4.5	Sorties d'impulsions	73
5	Création facile de la configuration des appareils.....	75
5.1	Chargement de la configuration d'une CPU connectée.....	76
5.2	Ajout d'une CPU à la configuration	78
5.3	Changer d'appareil.....	79
5.4	Ajout de modules à la configuration.....	80
5.5	Commande de configuration.....	81
5.6	Configuration du fonctionnement de la CPU et des modules	82
5.6.1	Le mémento système et le mémento de cadence mettent à disposition des fonctionnalités standard	84
5.7	Configuration de l'adresse IP de la CPU	87
5.8	La protection d'accès à la CPU ou au bloc de code est facile	89
5.8.1	Protection du savoir-faire	92
5.8.2	Protection contre la copie	93
6	Programmation facilitée	95
6.1	Conception aisée de votre programme.....	95
6.1.1	Utilisation d'OB pour l'organisation de votre programme utilisateur	97
6.1.2	Programmation aisée des tâches modulaires avec les FB et FC	100
6.1.3	Stockage aisément des données de programme dans des blocs de données	101
6.1.4	Création d'un nouveau bloc de code	102
6.1.5	Création de blocs de code réutilisables	103
6.1.6	Appel d'un bloc de code à partir d'un autre bloc de code	104
6.2	Langages de programmation faciles à utiliser	105
6.2.1	Schéma à contacts (CONT).....	105
6.2.2	Logigramme (LOG)	106
6.2.3	Vue d'ensemble du SCL	107
6.2.4	Éditeur de programme SCL	108
6.3	Les opérations puissantes facilitent la programmation.....	110

6.3.1	Mise à disposition des instructions de base dont vous avez besoin	110
6.3.2	Comparaison et transfert	113
6.3.3	Conversion	115
6.3.4	Les maths simplifiés avec l'opération Calculer	117
6.3.5	Fonctionnement des temporisations	118
6.3.6	Fonctionnement des compteurs	124
6.3.7	Modulation de largeur d'impulsion (PWM)	126
6.4	Enregistrements de données faciles à créer	127
6.5	Facilité de surveillance et d'essai de votre programme utilisateur	130
6.5.1	Tables de visualisation et tables de forçage permanent	130
6.5.2	Affichage de l'usage des références croisées	131
6.5.3	Structure d'appel permettant de constater la hiérarchie d'appel	132
6.5.4	Opérations de diagnostic pour contrôler le matériel	133
6.5.4.1	Lecture de l'état des DEL sur la CPU	133
6.5.4.2	Instructions pour la lecture de l'état de diagnostic des appareils	134
6.6	Compteur rapide (HSC)	135
6.6.1	Fonctionnement du compteur rapide	137
6.6.2	Configuration du HSC	144
7	Communication aisée entre les appareils	147
7.1	Création d'une liaison réseau	148
7.2	Options de communication	149
7.3	Liaisons de communication asynchrones V4.1	151
7.4	Instructions PROFINET et PROFIBUS	154
7.5	PROFINET	156
7.5.1	Communication ouverte (Open User Communication)	156
7.5.1.1	Mode ad hoc	157
7.5.1.2	ID de liaison pour les instructions Open User Communication	157
7.5.1.3	Paramètres pour la liaison PROFINET	161
7.5.2	Configuration du routage local/partenaire	163
7.6	PROFIBUS	166
7.6.1	Services de communication des CM PROFIBUS	167
7.6.2	Référence aux manuels utilisateurs des CM PROFIBUS	168
7.6.3	Ajout du module CM 1243-5 (maître DP) et d'un esclave DP	169
7.6.4	Affectation d'adresses PROFIBUS au module CM 1243-5 et à l'esclave DP	170
7.7	AS-i	172
7.7.1	Ajout du module maître AS-i CM 1243-2 et d'un esclave AS-i	173
7.7.2	Affectation d'une adresse AS-i à un esclave AS-i	174
7.8	Communication S7	177
7.8.1	Instructions GET et PUT	177
7.8.2	Création d'une liaison S7	178
7.8.3	Paramétrage de la liaison GET/PUT	179
7.9	GPRS	180
7.9.1	Connexion à un réseau GSM	180
7.9.2	Applications du CP 1242-7	182
7.9.3	Autres propriétés du CP-1242-7	183
7.9.4	Configuration et connexions électriques	183

7.9.5	Informations supplémentaires	184
7.9.6	Accessoires	184
7.9.7	Référence au manuel d'antenne GSM	185
7.9.8	Exemples de configuration pour telecontrol	186
7.10	Protocoles de communication PtP, USS et Modbus	191
7.10.1	Communication point à point	191
7.10.2	Utilisation des interfaces de communication série	193
7.10.3	Instructions point à point	194
7.10.4	Opérations USS	195
7.10.5	Opérations Modbus	197
8	Le PID, c'est simple	199
8.1	Insertion de l'instruction PID et de l'objet technologique	201
8.2	Instruction PID_Compact	203
8.3	Paramètre ErrorBits de l'instruction PID_Compact	207
8.4	Instruction PID_3Step	209
8.5	Paramètres ErrorBit de l'instruction PID_3Step	216
8.6	Instruction PID_Temp	218
8.6.1	Présentation	218
8.6.2	Fonctionnement du régulateur PID_Temp	222
8.6.3	Régulateurs en cascade	225
8.7	Paramètres ErrorBit de l'instruction PID_Temp	229
8.8	Configuration des contrôleurs PID_Compact et PID_3Step	231
8.9	Configuration du régulateur PID_Temp	234
8.10	Mise en service des régulateurs PID_Compact et PID_3Step	249
8.11	Mise en service du régulateur PID_Temp	251
9	Serveur Web pour une connectivité Internet facile	263
9.1	Pages Web standard faciles à utiliser	264
9.2	Contraintes qui peuvent affecter l'utilisation du serveur Web	267
9.3	Pages Web personnalisées faciles à créer	269
9.3.1	Pages Web "personnalisées" faciles à créer	269
9.3.2	Contraintes spécifiques aux pages Web personnalisées	271
9.3.3	Configuration d'une page Web personnalisée	272
9.3.4	Utilisation de l'opération WWW	273
10	La commande de mouvement est simple	275
10.1	Phasage	281
10.2	Configuration d'un générateur d'impulsions	284
10.3	Commande de mouvement	285
10.3.1	Configuration de l'axe	285
10.3.2	Mise en service	289
10.4	Régulation du mouvement	295
10.4.1	Configuration de l'axe	295

10.4.2	Mise en service	302
10.5	Configuration de la table de commande TO_CommandTable_PTO	308
10.6	Fonctionnement de la commande de mouvement pour le S7-1200.....	312
10.6.1	Sorties de la CPU pour la commande de mouvement	312
10.6.2	Fins de course matériels et logiciels pour la commande de mouvement.....	314
10.6.3	Référencement	318
10.6.3.1	Référencement de l'axe	318
10.6.3.2	Configuration des paramètres de référencement	320
10.6.3.3	Séquence pour le référencement actif.....	322
10.7	Instructions Motion Control	324
10.7.1	Présentation des instructions MC	324
10.7.2	Instruction MC_Power (Libérer/bloquer l'axe).....	325
10.7.3	Instruction MC_Reset (Confirmer l'erreur)	328
10.7.4	Instruction MC_Home (Référencer l'axe).....	329
10.7.5	Instruction MC_Halt (Pause de l'axe)	332
10.7.6	Instruction MC_MoveAbsolute (Positionnement absolu de l'axe).....	334
10.7.7	Instruction MC_MoveRelative (Positionnement relatif de l'axe)	336
10.7.8	Instruction MC_MoveVelocity (Déplacer l'axe à une vitesse prédéfinie)	338
10.7.9	Instruction MC_MoveJog (Déplacer l'axe en mode Manuel à vue)	341
10.7.10	Instruction MC_CommandTable (Exécuter des commandes d'axe en tant que séquence de mouvement)	344
10.7.11	Instruction MC_ChangeDynamic (Modifier les réglages dynamiques pour l'axe)	347
10.7.12	Instruction MC_WriteParam (écrire les paramètres de l'objet technologique)	349
10.7.13	Instruction MC_ReadParam (lire les paramètres de l'objet technologique)	351
11	Utilisation aisée des outils en ligne	353
11.1	Passage en ligne et connexion à une CPU	353
11.2	Interaction avec la CPU en ligne	354
11.3	Passage en ligne pour visualiser les valeurs dans la CPU	355
11.4	L'affichage de l'état du programme utilisateur est simple.....	356
11.5	Utilisation d'une table de visualisation pour surveiller la CPU.....	357
11.6	Utilisation de la table de forçage permanent	359
11.7	Acquisition des valeurs en ligne d'un DB pour redéfinir les valeurs initiales	362
11.8	Chargement des éléments du projet.....	363
11.9	Comparaison de CPU hors ligne et en ligne	364
11.10	Affichage des événements de diagnostic	365
11.11	Réglage de l'adresse IP et de l'heure	365
11.12	Restauration des réglages d'usine.....	366
11.13	Mise à jour du firmware.....	368
11.14	Chargement d'une adresse IP permanente dans une CPU en ligne.....	369
11.15	Utilisation de la "CPU non spécifiée" pour charger la configuration matérielle	370
11.16	Chargement dans la CPU à l'état MARCHE.....	371
11.16.1	Modification du programme à l'état "Marche"	373

11.17	Traçage et enregistrement de données CPU en fonctions de conditions de déclenchement.....	374
12	IO-Link est facile	375
12.1	Vue d'ensemble de la technologie IO-Link	375
12.2	Composants d'un système IO-Link	375
12.3	Après la mise sous tension	375
12.4	protocole IO-Link.....	376
12.5	Configuration dans le bus de terrain	376
12.6	IO-Link et le programme STEP 7	376
12.7	Le maître SM 1278 4xIO-Link.....	377
A	Caractéristiques techniques	381
A.1	Caractéristiques techniques d'ordre général	381
A.2	Modules CPU	392
A.3	Modules E/S numériques	397
A.3.1	Entrée/sortie TOR SB 1221 SB 1222 et SB 1223 (DI, DQ et DI/DQ).....	397
A.3.2	Entrée numérique (DI) SM 1221	400
A.3.3	Sortie numérique (DQ) SM 1222	401
A.3.4	Entrées/sorties TOR (DI / DQ) SM 1223	403
A.3.5	SM 1223 120/230 VAC input / Relay output	404
A.4	Caractéristiques pour les entrées et sorties numériques	405
A.4.1	Entrées TOR (DI) 24 V CC	405
A.4.2	Entrées TOR 120/230 V~	407
A.4.3	Sorties numériques (DQ)	408
A.5	Modules E/S analogiques	412
A.5.1	Entrée (AI) et sortie (AQ) analogiques SB 1231 et SB 1232	412
A.5.2	Entrée analogique (AI) SM 1231.....	413
A.5.3	Sortie analogique (AQ) SM 1232	413
A.5.4	Entrée-sortie analogique SM 1234 (AI/AQ)	413
A.5.5	Schémas de câblage pour SM 1231 (AI), SM 1232 (AQ) et SM 1234 (AI/AQ)	414
A.6	battery board BB 1297	415
A.7	Caractéristiques des E/S analogiques	416
A.7.1	Caractéristiques des entrées analogiques (CPU, SM et SB)	416
A.7.2	Plages de mesures d'entrée (AI) pour tension et courant	417
A.7.3	Réponse transitoire pour les entrées analogiques (AI)	419
A.7.4	Temps d'échantillonnage et temps d'actualisation pour les entrées analogiques	419
A.7.5	Caractéristiques des sorties analogiques	420
A.7.6	Plages de mesures de sortie (AQ) pour tension et courant	421
A.8	Modules RTD et Thermocouple	423
A.8.1	Caractéristiques RTD SB 1231 et TC SB 1231	424
A.8.2	Caractéristiques RTD SM 1231	425
A.8.3	Caractéristiques TC SM 1231	427
A.8.4	Caractéristiques d'entrée analogique pour RTD et TC (SM et SB)	428
A.8.5	Type de thermocouple	429
A.8.6	Sélection de filtres de thermocouple et de temps d'actualisation	430

A.8.7	Table de sélection type de capteur RTD	431
A.8.8	Sélection de filtres de thermocouple et de temps d'actualisation	432
A.9	Interfaces de communication	433
A.9.1	Maître/esclave PROFIBUS	433
A.9.1.1	ESCLAVE PROFIBUS DP CM 1242-5	433
A.9.1.2	Brochage de la prise subD du CM 1242-5	434
A.9.1.3	Maître PROFIBUS DP CM 1243-5	435
A.9.1.4	Le maître PROFIBUS (CM 1243-5) reçoit son alimentation 24 V- de la CPU	436
A.9.1.5	Brochage de la prise subD du CM 1243-5	437
A.9.2	CP GPRS	438
A.9.2.1	CP 1242-7 GPRS	438
A.9.2.2	Antenne GSM/GPRS ANT794-4MR	439
A.9.2.3	Antenne en nappe ANT794-3M	440
A.9.3	Teleservice (TS)	441
A.9.4	Communication RS485, RS232 et RS422	442
A.9.4.1	Caractéristiques CB 1241 RS485	442
A.9.4.2	Caractéristiques du module CM 1241 RS422/485	444
A.9.4.3	Caractéristiques du CM 1241 RS232	445
A.10	Modules technologiques	447
A.10.1	SM Maître SM 1278 4xIO-Link	447
A.10.1.1	Caractéristiques techniques du module d'entrées-sorties du Maître SM 1278 4xIO-Link	447
A.10.1.2	Schémas de câblage du module SM 1278 4 x IO-Link maître	449
A.11	Produits connexes	450
A.11.1	Module d'alimentation PM 1207	450
A.11.2	Module commutateur compact CSM 1277	450
A.11.3	Module CM CANopen	451
B	Remplacement d'une CPU V3.0 par une CPU V4.1	453
B.1	Remplacement d'une CPU V3.0 par une CPU V4.1	453
Index	459

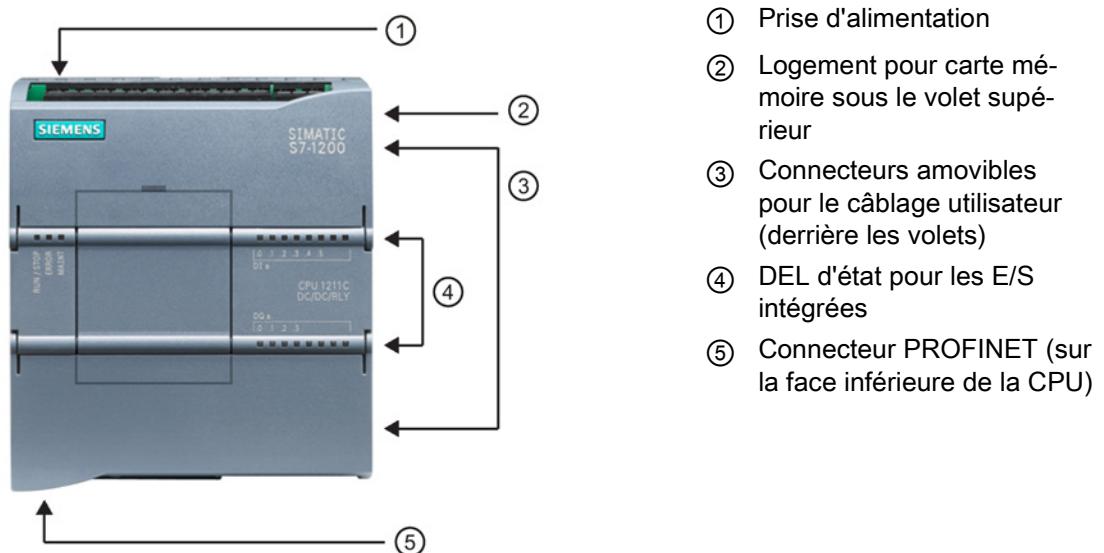
Introduction au S7-1200 puissant et souple

1.1 Introduction à l'automate S7-1200

Le contrôleur S7-1200 offre la souplesse et la puissance nécessaires pour commander une large gamme d'appareils afin de répondre à vos besoins en matière d'automatisation. Sa forme compacte, sa configuration souple et son important jeu d'instructions en font une solution idéale pour la commande d'applications très variées.

La CPU combine un microprocesseur, une alimentation intégrée, des circuits d'entrée et de sortie, un PROFINET intégré, des E/S rapides de commande de mouvement, ainsi que des entrées analogiques intégrées dans un boîtier compact en vue de créer un contrôleur puissant. Une fois que vous avez chargé votre programme, la CPU contient la logique nécessaire au contrôle et à la commande des appareils dans votre application. La CPU surveille les entrées et modifie les sorties conformément à la logique de votre programme utilisateur, qui peut contenir des instructions booléennes, des instructions de comptage, des instructions de temporisation, des instructions mathématiques complexes ainsi que des commandes pour communiquer avec d'autres appareils intelligents.

La CPU fournit un port PROFINET permettant de communiquer par le biais d'un réseau PROFINET. Des modules supplémentaires sont disponibles pour communiquer via les réseaux PROFIBUS, GPRS, RS485, RS232, IEC, DNP3 et WDC.



1.1 Introduction à l'automate S7-1200

Plusieurs fonctions de sécurité vous aident à protéger l'accès à la CPU et au programme de commande :

- Chaque CPU fournit une protection par mot de passe (Page 89) qui vous permet de configurer l'accès aux fonctions CPU.
- Vous pouvez utiliser la protection du savoir-faire (Page 92) (protection "know-how") pour masquer le code d'un bloc spécifique.
- Vous pouvez utiliser la protection contre la copie (Page 93) pour lier votre programme à une carte mémoire ou une CPU spécifique.

Tableau 1- 1 Comparaison des modèles de CPU

Caractéristique	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C			
Dimensions (mm)	90 x 100 x 75		110 x 100 x 75	130 x 100 x 75	150 x 100 x 75			
Mémoire utilisateur	de travail	50 Ko	75 Ko	100 Ko	125 Ko			
	de charge- ment	1 Mo		4 Mo				
	rémanente	10 Ko						
E/S intégrées lo- cales	TOR	6 entrées/4 sorties	8 entrées/6 sorties	14 entrées/10 sorties				
	Analo- giques	2 entrées			2 entrées/2 sorties			
Taille de la mé- moire image	Entrées (I)	1024 octets						
	Sorties (Q)	1024 octets						
Mémentos (M)	4096 octets		8192 octets					
Modules d'entrées-sorties (SM) pour extension	Aucun	2	8					
Signal Board (SB), Battery Board (BB) ou Communication Board (CB)	1							
Module de communication (CM) (extension vers la gauche)	3							
Compteurs rapides	Total	Jusqu'à 6 configurés pour utiliser des entrées intégrées ou SB						
	1 MHz	-			Ib.2 à Ib.5			
	100/180 kHz	Ia.0 à Ia.5						
	30/120 kHz	--	Ia.6 à Ia.7	Ia.6 à Ib.5	Ia.6 à Ib.1			
	200 kHz ³							
Sorties d'impul- sions ²	Total	Jusqu'à 4 configurées pour utiliser des sorties intégrées ou SB						
	1 MHz	--			Qa.0 à Qa.3			
	100 kHz	Qa.0 à Qa.3			Qa.4 à Qb.1			
	20 kHz	--	Qa.4 à Qa.5	Qa.4 à Qb.	--			
Carte mémoire	Carte mémoire SIMATIC (facultative)							
Durée de conservation de l'hor- loge temps réel	20 jours typ./12 jours min. à 40 °C (supercondensateur sans maintenance)							
Port de communication Ethernet PROFINET	1		2					

Caractéristique	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C
Vitesse d'exécution des instructions mathématiques sur réels	2,3 µs/instruction				
Vitesse d'exécution des instructions booléennes	0,08 µs/instruction				

- ¹ La vitesse plus lente s'applique lorsque le HSC est configuré pour fonctionner en quadrature de phase.
- ² Pour les modèles de CPU avec sorties relais, vous devez installer un Signal Board (SB) TOR pour utiliser les sorties d'impulsions.
- ³ Jusqu'à 200 kHz sont disponibles avec le SB 1221 DI x 24 VDC 200 kHz et le SB 1221 DI 4 x 5 VDC 200 kHz.

Les différents modèles de CPU disposent de fonctions et éléments divers qui vous aident à créer des solutions efficaces pour vos diverses applications. Reportez-vous aux caractéristiques techniques (Page 381) pour des informations détaillées sur une CPU spécifique.

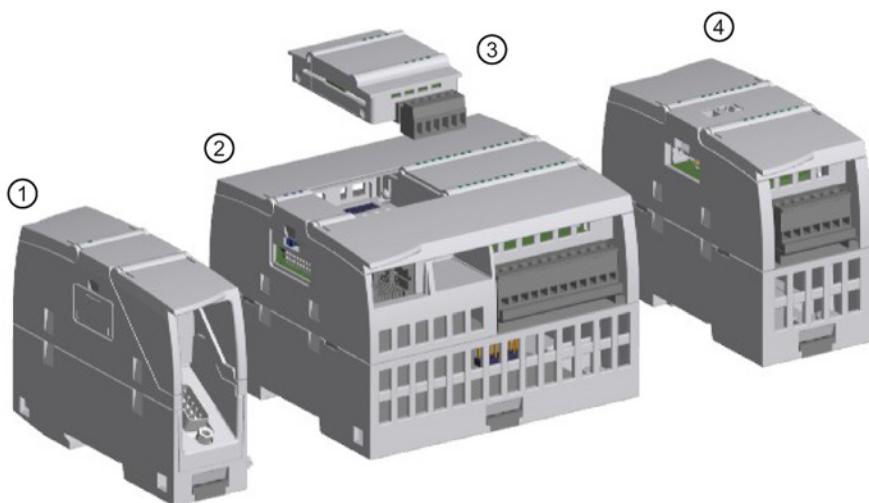
Tableau 1- 2 Blocs, temporisations et compteurs pris en charge par le S7-1200

Elément	Description	
Blocs	Type	OB, FB, FC, DB
	Taille	50 Ko (CPU 1211C) 75 Ko (CPU 1212C) 100 Ko (CPU 1214C) 125 Ko (CPU 1215C) 150 Ko (CPU 1217C)
	Quantité	Jusqu'à 1024 blocs au total (OB + FB + FC + DB)
	Profondeur d'imbrication	16 en cas d'appel depuis l'OB de cycle de programme ou de démarrage ; 6 depuis n'importe quel OB d'alarme
	Visualisation	Il est possible de visualiser simultanément l'état de 2 blocs de code.
OB	Cycle de programme	Multiple
	Mise en route	Multiple
	Alarmes temporisées	4 (1 par événement)
	Alarmes cycliques	4 (1 par événement)
	Alarmes de processus	50 (1 par événement)
	Alarmes d'erreur de temps	1
	Alarmes de diagnostic	1
	Débrouillage ou enfichage de modules	1
	Défaillance du châssis ou de la station	1
	Heure	Multiple
	Etat	1
	Mettre à jour	1
	Profil	1
Temporisations	Type	CEI
	Quantité	Limité uniquement par la taille de la mémoire
	Stockage	Structure dans un DB, 16 octets par temporisation

Elément	Description	
Compteurs	Type	CEI
	Quantité	Limité uniquement par la taille de la mémoire
	Stockage	Structure dans un DB, la taille dépend du type de compteur <ul style="list-style-type: none">• SInt, USInt : 3 octets• Int, UInt : 6 octets• DInt, UDInt : 12 octets

1.2 Possibilités d'extension de la CPU

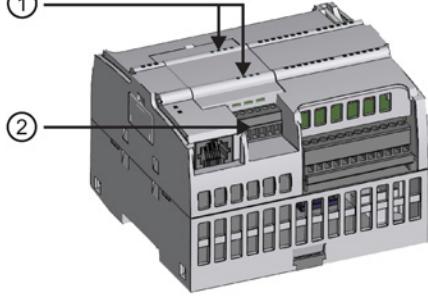
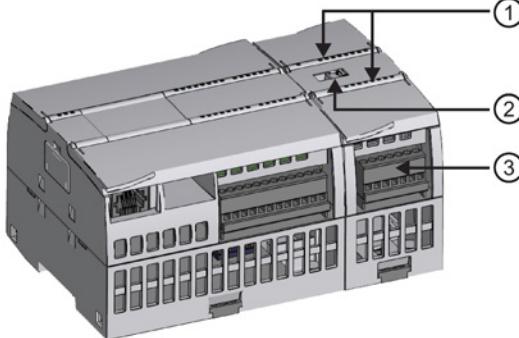
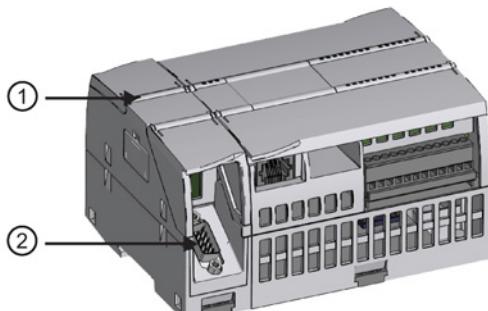
La gamme S7-1200 offre divers modules et cartes enfichables pour accroître les capacités de la CPU avec des E/S supplémentaires ou d'autres protocoles de communication. Référez-vous aux caractéristiques techniques (Page 381) pour des informations détaillées sur un module spécifique.



- ① Module de communication (CM) ou processeur de communication (CP)
- ② CPU (CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C, CPU 1217C)
- ③ Signal board (SB) (SB TOR, SB analogique), communication board (CB) ou Battery Board (BB)CPU (CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C, CPU 1217C)
- ④ Module d'entrées-sorties (SM) (SM TOR, SM analogique, SM thermocouple, SM RTD, SM technologique)

1.3 Modules S7-1200

Tableau 1- 3 Modules d'extension S7-1200

Type de module	Description
<p>La CPU prend en charge une carte d'extension enfichable :</p> <ul style="list-style-type: none"> • Un Signal Board (SB) fournit des E/S supplémentaires pour votre CPU. Le SB se raccorde à l'avant de la CPU. • Un Communication Board (CB) vous permet d'ajouter un autre port de communication à votre CPU. • Un Battery Board (BB) permet une sauvegarde à long terme de l'horloge temps réel. 	 <p>① DEL d'état sur le Signal Board ② Connecteur amovible pour le câblage utilisateur</p>
<p>Les modules d'entrées-sorties (SM) permettent d'ajouter des fonctionnalités à la CPU. Les SM se raccordent sur le côté droit de la CPU.</p> <ul style="list-style-type: none"> • E/S TOR • E/S analogiques • RTD et Thermocouple • SM 1278 IO-Link maître 	 <p>① DEL d'état ② Languette coulissante du connecteur de bus ③ Connecteur amovible pour le câblage utilisateur</p>
<p>Les modules de communication (CM) et les processeurs de communication (CP) ajoutent des options de communication à la CPU, telles que la connectivité PROFIBUS ou RS232/RS485 (pour PtP, Modbus ou USS) ou le maître AS-i.</p> <p>Un CP offre la possibilité d'autres types de communication, par exemple la connexion de la CPU par le biais d'un réseau GPRS, IEC, DNP3 ou WDC.</p> <ul style="list-style-type: none"> • La CPU accepte jusqu'à trois CM ou CP. 	

Type de module	Description
• Chaque CM ou CP se raccorde sur le côté gauche de la CPU (ou sur le côté gauche d'un autre CM ou CP).	① DEL d'état
	② Connecteur de communication

1.4 Tableaux IHM basiques

Les pupitres SIMATIC HMI Basic Panels fournissent des appareils à écran tactile pour des tâches de base de commande et de surveillance d'opérateur. Tous les tableaux ont un indice de protection de IP65 et sont certifiés CE, UL, cULus et NEMA 4x.

Les pupitres HMI Basic Panels disponibles sont décrits ci-dessous :

- KTP400 Basic : écran tactile 4" avec 4 touches configurables, une résolution de 480 x 272 et 800 variables
- KTP700 Basic : écran tactile 7" avec 8 touches configurables, une résolution de 800 x 480 et 800 variables
- KTP700 Basic DP : écran tactile 7" avec 8 touches configurables, une résolution de 800 x 480 et 800 variables
- KTP900 Basic : écran tactile 9" avec 8 touches configurables, une résolution de 800 x 480 et 800 variables
- KTP1200 Basic : écran tactile 12" avec 10 touches configurables, une résolution de 800 x 480 et 800 variables
- KTP 1200 Basic DP : écran tactile 12" avec 10 touches configurables, une résolution de 800 x 400 et 800 variables

1.5 Dimensions de montage et dégagement requis

L'API S7-1200 est conçu pour une installation facile. Quel que soit le type de montage, sur panneau ou sur profilé DIN standard, la dimension compacte optimise l'espace disponible.

Pour les exigences et directives d'installation spécifiques, référez-vous au *Manuel système de l'automate programmable S7-1200*.

CPU 1211C, CPU 1212C, CPU 1214C
(measurements in mm)

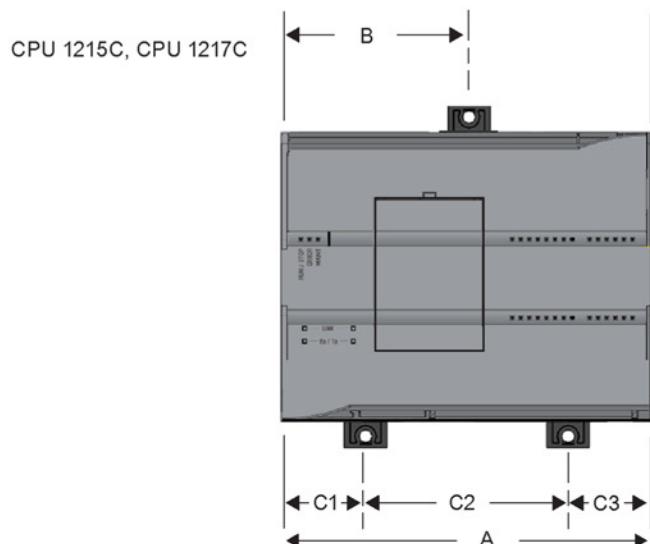
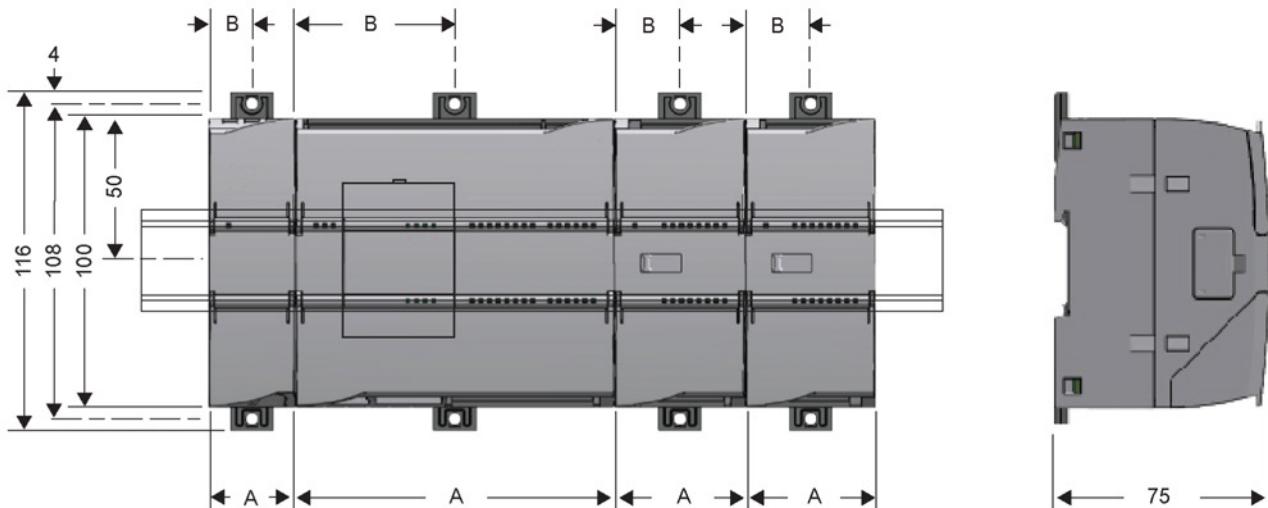


Tableau 1- 4 Dimensions de montage (mm)

Appareils S7-1200		Largeur A (mm)	Largeur B (mm)	Largeur C (mm)
CPU	CPU 1211C et CPU 1212C	90	45	--
	CPU 1214C	110	55	--
	CPU 1215C	130	65 (haut)	Bas : C1 : 32.5 C2 : 65 C3 : 32.5
	CPU 1217C	150	75	Bas : C1 : 37.5 C2 : 75 C3 : 37.5
Modules d'entrées-sorties	TOR, 8 et 16 entrées/sorties Analogiques, 2, 4 et 8 entrées/sorties Thermocouple, 4 et 8 entrées/sorties RTD, 4 entrées/sorties SM 1278 maître IO-Link	45	22.5	--
	TOR, 8 sorties Relais (inverseur)	70	35	--
	Analogiques, 16 entrées/sorties RTD, 8 entrées/sorties	70	35	--
Interfaces de communication	CM 1241 RS232 et CM 1241 RS422/485	30	15	--
	CM 1243-5 maître PROFIBUS et CM 1242-5 esclave PROFIBUS			
	CM 1242-2 maître AS-i			
	CP 1242-7 GPRS V2			
	CP 1243-7 LTE-EU			
	CP 1243-1 DNP3			
	CP 1243-1 IEC			
	CP 1243-1			
	CP1243-1 PCC			
	CP 1243-8 ST7			
	RF120C			
	Adaptateur TS (TeleService) Adapter IE Advanced ¹			
	Adaptateur TS (TeleService) Adapter IE Basic ¹	30	15	--
	Adaptateur TS Adapter	30	15	--
	Module TS			

¹ Avant d'installer l'adaptateur TS (TeleService) Adapter IE Advanced ou IE Basic, vous devez d'abord connecter l'adaptateur TS Adapter et un module TS. La largeur totale ("largeur A") est de 60 mm.

Chaque CPU, SM, CM et CP peut être indifféremment monté sur un profilé support ou encastré dans un panneau. Servez-vous des barrettes de fixation sur le module pour fixer l'appareil sur le profilé support. Ces barrettes s'encliquettent également en position sortie afin de fournir des points de vissage pour monter l'unité directement sur un panneau. Le diamètre intérieur de l'alésage pour les barrettes de fixation de l'appareil est de 4,3 mm.

Une zone thermique de 25 mm doit également être ménagée au-dessus et en dessous de l'unité pour la libre circulation de l'air.

Le matériel S7-1200 est conçu pour être facile à installer. Vous pouvez monter l'automate S7-1200 sur un panneau ou sur un profilé support et l'orienter horizontalement ou verticalement. La petite taille du S7-1200 permet une optimisation de l'espace.

Les CPU de sécurité du S7-1200 ne prennent pas en charge les E/S de sécurité distribuées par PROFIBUS ou PROFINET.

Les normes relatives au matériel électrique classent le système SIMATIC S7-1200 comme équipement ouvert. Vous devez installer le S7-1200 dans un boîtier, une armoire ou une salle électrique auxquels seules les personnes autorisées doivent avoir accès.

Le S7-1200 doit être installé dans un environnement sec. On considère que les circuits TBTS/TBTP offrent une protection contre les chocs électriques dans des endroits secs.

L'installation doit offrir une protection mécanique et environnementale homologuée pour un équipement ouvert dans la catégorie de votre emplacement selon les codes électriques et de construction applicables.

Une contamination conductrice occasionnée par la poussière, l'humidité ou la pollution atmosphérique peut entraîner des dysfonctionnements et des défauts électriques dans l'API.

Si vous placez l'API dans une zone susceptible de renfermer une pollution conductrice, l'API doit être protégé dans une enceinte ayant un indice de protection approprié. L'indice IP54 est généralement utilisé pour les enceintes de matériel électrique dans des environnements souillés et il se peut qu'il soit approprié pour votre application.

 ATTENTION
<p>Une installation inappropriée du S7-1200 peut provoquer des défauts électriques ou un fonctionnement inattendu des machines.</p>
<p>Les défauts électriques ou un fonctionnement inattendu des machines peuvent entraîner la mort, des blessures graves et des dommages matériels importants.</p>
<p>Il convient de respecter toutes les instructions d'installation et de maintien d'un environnement de fonctionnement approprié afin d'assurer le fonctionnement sûr de l'équipement.</p>

Ne placez pas les appareils S7-1200 à proximité de sources de chaleur, de haute tension et de bruit électrique

En règle générale pour la mise en place des appareils de votre système, séparez toujours les appareils générant une tension élevée et un bruit électrique important des appareils de type logique basse tension, tels que le S7-1200.

Lors de l'implantation du S7-1200 dans votre panneau, tenez compte des appareils sources de chaleur et placez les appareils de type électronique dans les zones plus fraîches de votre armoire. En effet, réduire l'exposition aux températures élevées allonge la durée de vie des appareils électroniques.

Tenez également compte de l'acheminement du câblage pour les appareils dans le panneau. Evitez de placer les câbles de signaux et de communication de faible tension dans la même goulotte que le câblage d'alimentation en courant alternatif et le câblage pour courant continu à commutation rapide et haute énergie.

Ménagez un dégagement adéquat pour le refroidissement et le câblage

Les appareils S7-1200 sont conçus pour un refroidissement par convection naturelle. Pour que le refroidissement se fasse correctement, vous devez laisser un espace libre d'au moins 25 mm au-dessus et en dessous des appareils. Vous devez également avoir une profondeur d'au moins 25 mm entre l'avant des modules et l'intérieur de l'enceinte.



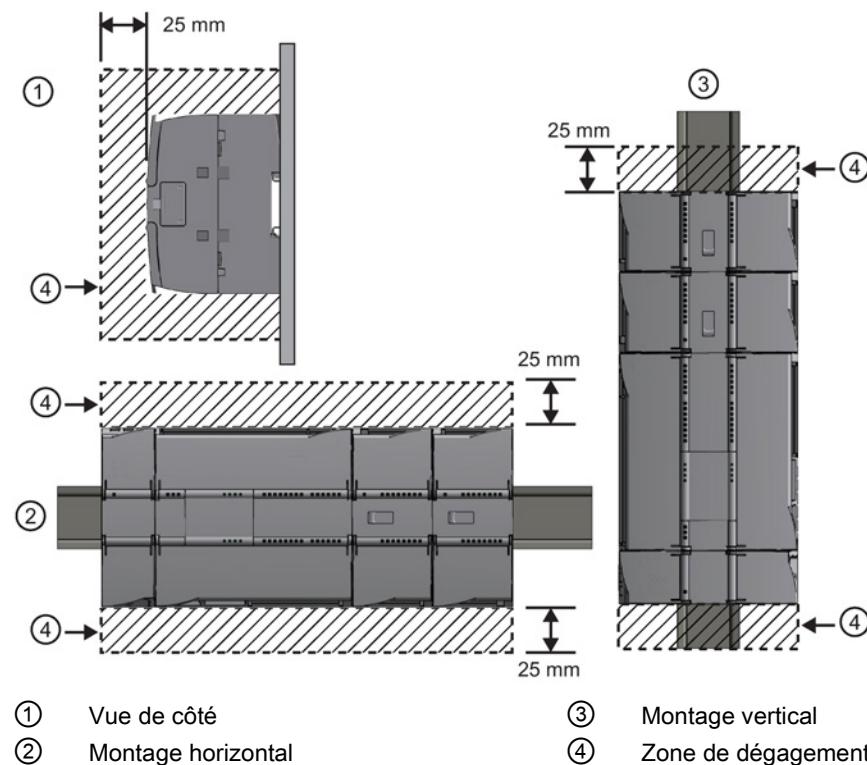
PRUDENCE

En cas de montage vertical, la température ambiante maximale autorisée est diminuée de 10 °C.

Orientez un système S7-1200 monté verticalement comme illustré dans la figure suivante.

Assurez-vous que le système S7-1200 est monté correctement.

Lorsque vous planifiez la disposition de votre système S7-1200, réservez suffisamment d'espace pour le câblage et les connexions de câbles de communication.



ATTENTION

L'installation ou le démontage du S7-1200 ou d'équipements reliés alors qu'ils sont sous tension peut provoquer un choc électrique ou un fonctionnement imprévisible du matériel.

La non-désactivation de l'alimentation complète du S7-1200 et des appareils qui y sont reliés pendant les procédures d'installation ou de désinstallation peut entraîner la mort, des blessures graves ou des dommages matériels importants en raison du choc électrique ou du fonctionnement imprévisible du matériel.

Prenez toujours toutes les mesures de sécurité nécessaires et assurez-vous que l'alimentation de l'automate S7-1200 est coupée avant de tenter d'installer ou de démonter des CPU S7-1200 ou des matériels reliés.

Assurez-vous toujours, lorsque vous remplacez ou installez un appareil S7-1200, que vous utilisez le bon module ou un appareil équivalent.

 **ATTENTION**

La mauvaise installation d'un module S7-1200 peut entraîner un fonctionnement imprévisible du programme dans le S7-1200.

Le remplacement d'un appareil S7-1200 par un autre modèle, sans respecter la bonne orientation et l'ordre initial, peut entraîner la mort, des blessures graves ou des dommages matériels importants en raison d'un fonctionnement imprévisible de l'équipement.

Remplacez toujours un appareil S7-1200 par le même modèle et assurez-vous que vous l'orientez et le positionnez correctement.

1.6 Nouvelles fonctions

Cette version comporte les nouvelles fonctions suivantes :

- Vous pouvez désormais mettre en œuvre la sécurité fonctionnelle, à l'aide du matériel et du firmware des CPU et des modules d'entrées-sorties (SM) de sécurité du S7-1200, conjointement avec le programme de sécurité téléchargé par le logiciel (ES). Pour plus d'informations, référez-vous au Manuel de sécurité fonctionnelle S7-1200 (<http://support.automation.siemens.com/WW/view/en/104547552>).
- Simulation des CPU S7-1200 avec une version de firmware V4.0 et supérieure : S7-PLCSIM V13 SP1 vous permet de tester vos programmes API sur une API simulée sans recourir à du matériel réel. S7-PLCSIM est une application installée séparément qui fonctionne en combinaison avec STEP 7 dans le TIA Portal. Vous pouvez configurer votre API et tout module associé dans STEP 7, programmer votre logique d'application, puis charger la configuration matérielle et le programme dans S7-PLCSIM. Vous pouvez ensuite utiliser les outils de S7-PLCSIM pour simuler et tester votre programme. Voir l'aide en ligne de S7-PLCSIM pour obtenir la documentation complète. À noter que vous ne pouvez pas simuler des CPU de sécurité.
- Commande de configuration (gestion des options) (Page 81) : vous pouvez configurer le matériel pour une configuration maximum de la machine comprenant les modules que vous pourriez ne pas utiliser en réalité pendant le fonctionnement. La configuration et la conception de ces modules flexibles est nouvelle avec cette version de STEP 7 et du S7-1200. Les modules que vous concevez ainsi ne provoqueront pas de situations d'erreur s'ils sont absents.
- Le serveur Web (Page 263) accepte maintenant un accès à travers l'adresse IP des modules (du processeur de communication) sélectionnés dans le châssis local ainsi qu'à travers l'adresse IP de la CPU S7-1200.
- Fonctionnalité de mouvement améliorée :
 - Connexions analogiques et PROFIdrive
 - Modulo et paramètres étendus de boucle de commande
- Mesure de période à l'aide de compteurs rapides (HSC) (Page 135)

- Améliorations des performances du compilateur SCL
- Liaison dynamique de protection contre la copie (Page 93) pour des blocs de programmes avec mot de passe obligatoire
- Fonctionnalité de PROFINET améliorée, y compris l'assistance pour les dispositifs partagés.
- Nouvelles instructions de programmation :
 - EQ_Type, NE_Type, EQ_ElemType, NE_ElemType
 - IS_NULL, NOT_NULL
 - IS_ARRAY
 - Deserialize, Serialize
 - VariantGet, VariantPut, CountOfElements
 - Variant_to_DB_Any, DB_Any_To_Variant
 - GET_IM_DATA
 - RUNTIME
 - GEO2LOG, IO2MOD
 - ReadLittle, WriteLittle, ReadBig, WriteBig (SCL uniquement)
 - T_RESET, T_DIAG et TMAIL_C
 - PID_Temp
 - Nouvelles instructions Modbus (Page 197)
 - Nouvelles instructions point à point (PtP) (Page 194)
 - Nouvelles instructions USS (Page 195)

Nouveaux modules pour le S7-1200

De nouveaux modules augmentent la puissance de la CPU du S7-1200 et offrent la souplesse nécessaire à vos besoins en automatisation :

- Modules industriels de communication de commande distants : vous pouvez utiliser les CPU comme modules de communication avec la CPU S7-1200 V4.1.
- CPU de sécurité et E/S : Il y a quatre CPU de sécurité et trois modules d'entrées-sorties (SM) de sécurité utilisés conjointement avec le S7-1200 V4.1 ou une version ultérieure :
 - CPU 1214FC DC/DC/DC (6ES7 214-1AF40-0XB0)
 - CPU 1214FC DC/DC/RLY (6ES7 214-1HF40-0XB0)
 - CPU 1215FC DC/DC/DC (6ES7 215-1AF40-0XB0)
 - CPU 1215FC DC/DC/RLY (6ES7 215-1HF40-0XB0)
 - SM 1226 F-DI 16 x 24 VDC (6ES7 226-6BA32-0XB0)
 - SM 1226 F-DQ 4 x 24 VDC (6ES7 226-6DA32-0XB0)
 - SM 1226 F-DQ 2 x Relay (6ES7 226-6RA32-0XB0)

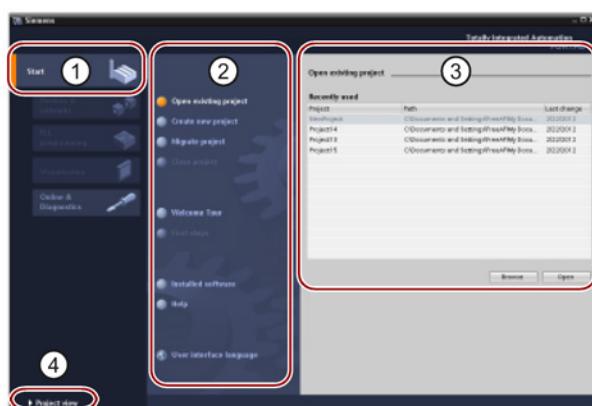
Vous pouvez utiliser les modules d'entrées-sorties (SM), les modules de communication (CM) et les signal boards (SB) standard du S7-1200 dans le même système avec les SM de sécurité pour compléter vos fonctions de commande d'application qui ne nécessitent pas de caractéristiques assignées en matière de sécurité fonctionnelle. Les SM standard qui sont acceptés pour une utilisation avec les SM de sécurité ont les références (6ES7 ---32 0XB0) ou supérieures.

Remplacer votre CPU V3.0 par une CPU V4.1

Si vous remplacez une CPU S7-1200 V3.0 par une CPU S7-1200 V4.1, prenez connaissance des différences (Page 453) documentées entre ces versions et des actions utilisateur requises.

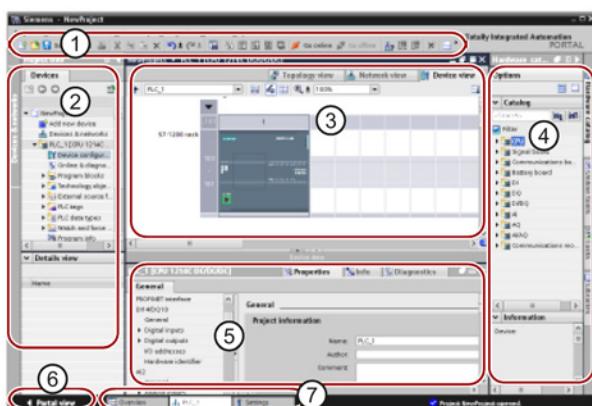
STEP 7 facilite le travail

STEP 7 offre un environnement convivial pour développer la logique du contrôleur, configurer la visualisation IHM et établir la communication réseau. Pour permettre d'augmenter votre productivité, STEP 7 offre deux vues différentes du projet : un ensemble orienté tâche de portails qui sont organisés selon la fonctionnalité des outils (vue du portail) et une vue orientée projet des éléments dans le projet (vue du projet). Choisissez la vue qui permet un travail le plus efficace possible. Avec un simple clic, vous pouvez faire le va-et-vient entre la vue du portail et la vue du projet.



Vue du portail

- ① Portails des différentes tâches
- ② Tâches du portail sélectionné
- ③ Panneau de sélection de l'action
- ④ Bascule dans la vue du projet



Vue du projet

- ① Menus et barre d'outils
- ② Navigateur du projet
- ③ Zone de travail
- ④ Task Cards
- ⑤ Fenêtre d'inspection
- ⑥ Bascule dans la vue du portail
- ⑦ Barre d'édition

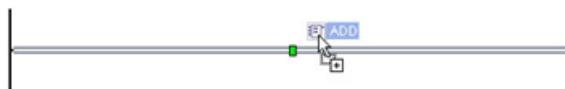
Comme tous ces composants sont regroupés à un endroit, vous pouvez facilement accéder à chaque élément de votre projet. La fenêtre d'inspection montre, par exemple, les propriétés et informations de l'objet que vous avez sélectionné dans la zone de travail. Lorsque vous sélectionnez différents objets, la fenêtre d'inspection affiche les propriétés que vous pouvez configurer. La fenêtre d'inspection contient des onglets vous permettant de voir les informations de diagnostic et autres messages.

2.1 Insertion facile d'opérations dans votre programme utilisateur

En affichant tous les éditeurs ouverts, la barre d'édition vous permet de travailler plus rapidement et efficacement. Pour basculer d'un éditeur ouvert à un autre, il suffit de cliquer sur l'éditeur correspondant. Vous pouvez également aligner deux éditeurs verticalement ou horizontalement. Cette fonction vous permet d'utiliser la fonction glisser-déplacer entre les éditeurs.

2.1 Insertion facile d'opérations dans votre programme utilisateur

STEP 7 fournit des Task Cards qui contiennent les instructions pour votre programme. Les opérations sont regroupées selon leur fonction.

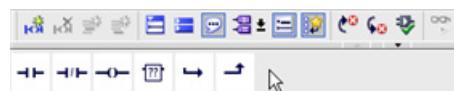


Pour créer votre programme, vous amenez les opérations de la Task Card dans un réseau.

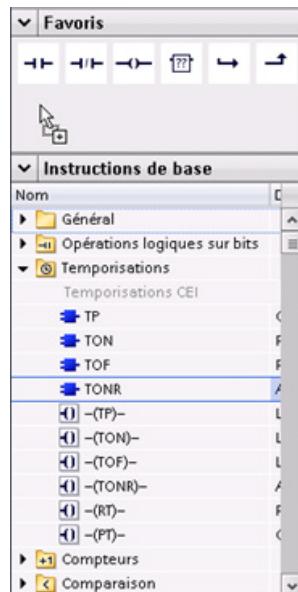


2.2 Accès facile à vos opérations favorites à partir d'une barre d'outils

STEP 7 fournit une barre d'outils "Favoris" pour permettre un accès rapide aux instructions que vous utilisez fréquemment. Effectuez un clic simple sur l'icône de l'opération à insérer dans votre réseau !



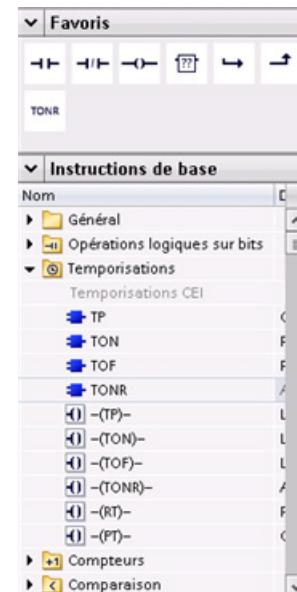
(Pour les "Favoris" dans l'arborescence d'instructions, double-cliquez sur l'icône)



Vous pouvez facilement personnaliser les "Favoris" en ajoutant de nouvelles opérations.

Il vous suffit de glisser-déplacer une opération sur vos "Favoris".

Un simple clic permet alors d'accéder à l'opération !



2.3 Il est facile d'ajouter des entrées ou sorties aux opérations CONT et LOG.

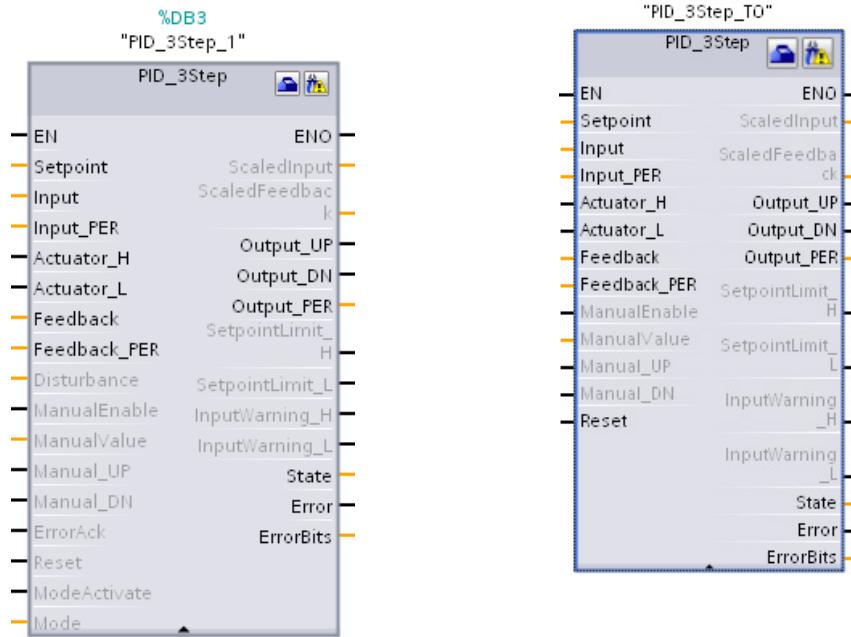


Certaines instructions vous permettent de créer des entrées ou sorties supplémentaires.

- Pour ajouter une entrée ou une sortie, cliquez sur l'icône de création ou cliquez avec le bouton droit de la souris sur la ligne de l'un des paramètres IN ou OUT existants et sélectionnez la commande "Insérer entrée".
- Pour supprimer une entrée ou une sortie, cliquez avec le bouton droit de la souris sur la ligne de l'un des paramètres IN ou OUT existants (lorsqu'il y a plus d'entrées que les deux entrées d'origine) et sélectionnez la commande "Supprimer".

2.4 Instructions extensibles

Quelques-unes des instructions les plus complexes sont extensibles, n'affichant que les entrées et sorties importantes. Pour afficher toutes les entrées et sorties, cliquez sur la flèche au bas de l'instruction.



2.5

Changement facile de l'état de fonctionnement de la CPU

La CPU ne possède pas de commutateur physique pour changer l'état de fonctionnement (ARRET ou MARCHE).

Cliquez sur les boutons "Démarrer CPU" ou "Arrêter CPU" de la barre d'outils pour changer l'état de fonctionnement de la CPU.



Lorsque vous configurez la CPU dans les paramètres de configuration, vous configurez le comportement au démarrage dans les propriétés de la CPU (Page 82).

Le portail "En ligne & diagnostic" comprend un panneau de commande pour modifier l'état de fonctionnement de la CPU en ligne. Pour pouvoir utiliser le panneau de commande de la CPU, vous devez être connecté en ligne à la CPU. La Task Card "Outils en ligne" affiche un panneau de commande indiquant l'état de fonctionnement de la CPU en ligne. Ce panneau de commande vous permet également de changer l'état de fonctionnement de la CPU en ligne.



Servez-vous du bouton sur le panneau de commande pour changer l'état de fonctionnement (ARRET ou MARCHE). Le panneau de commande dispose également d'un bouton MRES pour réinitialiser la mémoire.

La couleur de l'indicateur MARCHE/ARRET signale l'état de fonctionnement en cours de la CPU. Le jaune correspond à l'état ARRET, le vert à l'état MARCHE.

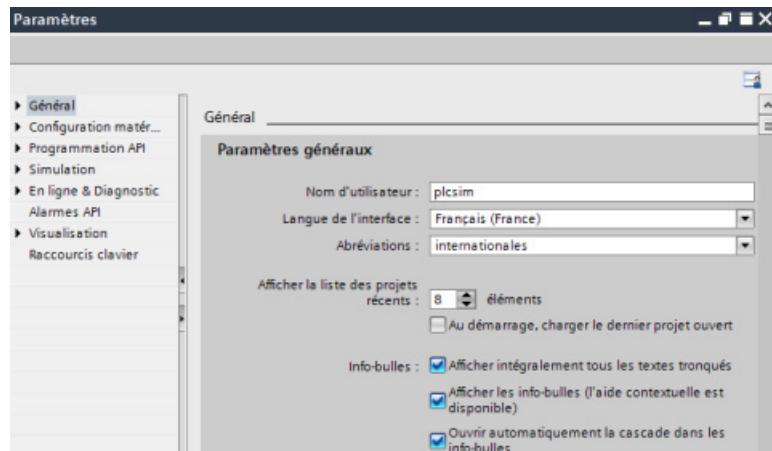
Dans la configuration de l'appareil de STEP 7, vous pouvez également configurer l'état de fonctionnement par défaut à la mise sous tension.

2.6

Il est facile de modifier l'apparence et la configuration de STEP 7

Vous pouvez sélectionner divers paramètres, tels que l'apparence de l'interface, la langue ou le dossier d'enregistrement de votre travail.

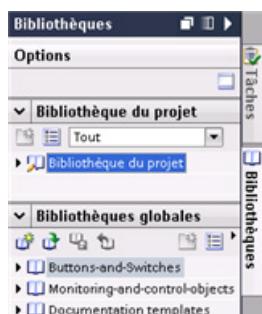
Pour modifier ces paramètres, sélectionnez la commande "Paramètres" dans le menu "Outils".



2.7

Accès aisément aux bibliothèques du projet et bibliothèques globales

Les bibliothèques globales et bibliothèques du projet vous permettent de réutiliser les objets stockés dans un projet ou multiprojet. Vous pouvez, par exemple, créer des modèles de blocs afin de les utiliser dans différents projets et de les adapter aux exigences spécifiques de votre tâche d'automatisation. Vous pouvez stocker divers objets dans les bibliothèques, comme p. ex. des FC, FB, DB, configurations d'appareils, types de données, tables de visualisation, vues du processus et faceplates. Vous pouvez également enregistrer les composants de l'appareil IHM dans votre projet.



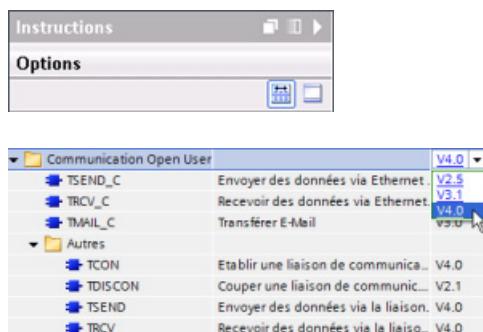
Chaque projet dispose d'une bibliothèque de projet permettant de stocker les objets à utiliser plus d'une fois dans le projet. Cette bibliothèque de projet fait partie du projet. En ouvrant ou fermant le projet, vous ouvrez ou fermez également la bibliothèque du projet et en enregistrant le projet, vous enregistrez également toute modification dans la bibliothèque du projet.

Vous pouvez créer votre propre bibliothèque globale afin d'y stocker les objets que vous souhaitez rendre disponible pour une utilisation dans l'autres projets. Lorsque vous créez une nouvelle bibliothèque globale, vous l'enregistrez à une adresse sur votre ordinateur ou réseau.

2.8

Sélection aisément de la version d'une instruction

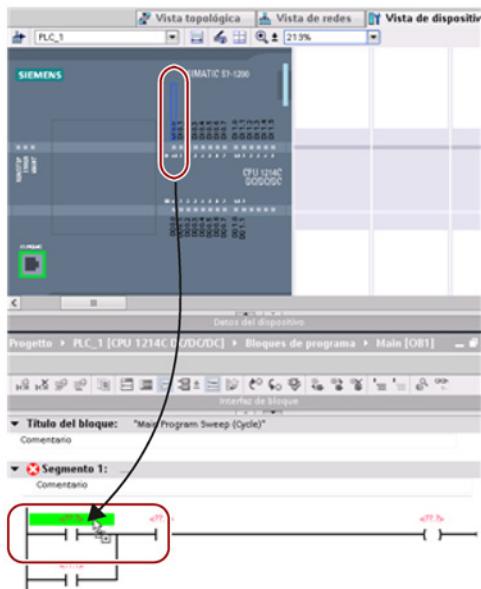
Les cycles de développement et de mise à disposition pour certains jeux d'instructions (tels que Modbus, PID et commande de mouvement) ont créé de multiples versions disponibles de ces instructions. Pour vous aider à garantir la compatibilité et la migration avec des projets plus anciens, STEP 7 vous permet de choisir la version d'une instruction à insérer dans votre programme utilisateur.



Dans la Task Card d'arborescence d'instructions, cliquez sur l'icône pour activer les en-têtes et colonnes de l'arborescence d'instructions.

Pour changer la version de l'instruction, sélectionnez la version appropriée dans la liste déroulante.

2.9 Glisser-déplacer facilement entre les éditeurs

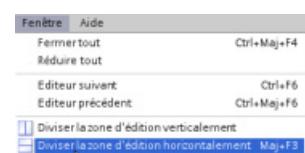


Pour afficher simultanément deux éditeurs, choisissez les commandes de menu "Fractionner éditeur" ou sélectionnez les boutons correspondants dans la barre d'outils.

Pour vous aider à réaliser des tâches rapidement et facilement, STEP 7 vous permet de faire glisser des éléments d'un éditeur à un autre. Vous pouvez, par exemple, amener une entrée de la CPU sur l'adresse d'une opération dans votre programme utilisateur.

Vous devez effectuer un agrandissement d'au moins 200 % pour pouvoir sélectionner les entrées ou les sorties de la CPU.

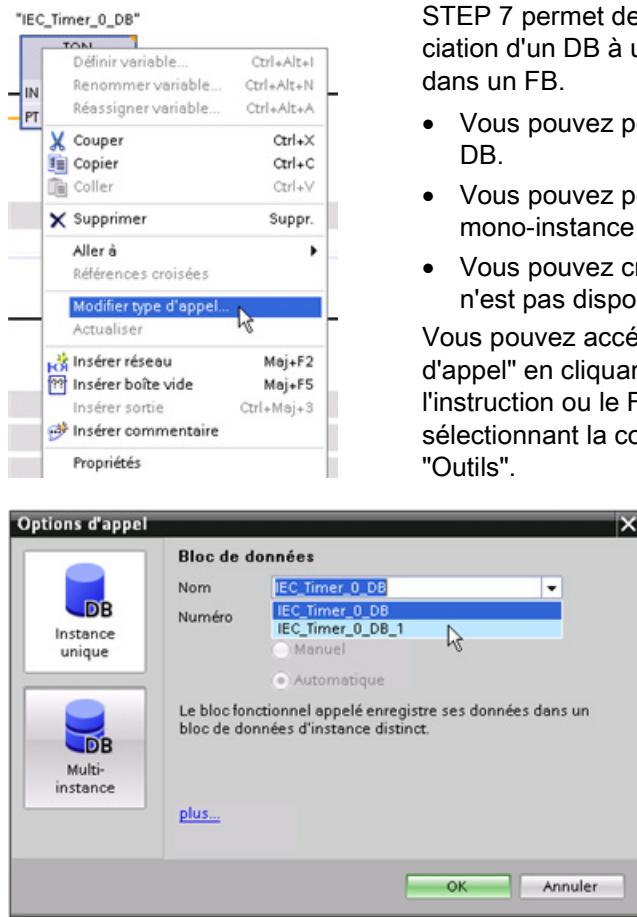
Notez que les noms des variables sont affichés non seulement dans la table des variables de l'API, mais également dans la CPU.



Pour basculer entre les éditeurs ouverts, cliquez sur les icônes correspondantes dans la barre d'édition.



2.10 Modification du type d'appel pour un DB



STEP 7 permet de créer et de changer aisément l'association d'un DB à une instruction ou à un FB se trouvant dans un FB.

- Vous pouvez permuter l'association entre différents DB.
- Vous pouvez permuter l'association entre un DB mono-instance et un DB multi-instance.
- Vous pouvez créer un DB d'instance (s'il manque ou n'est pas disponible).

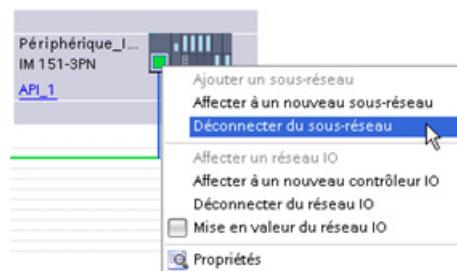
Vous pouvez accéder à la commande "Modifier type d'appel" en cliquant avec le bouton droit de la souris sur l'instruction ou le FB dans l'éditeur de programmes ou en sélectionnant la commande "Appel de bloc" dans le menu "Outils".

La boîte de dialogue "Options d'appel" vous permet de sélectionner un DB mono-instance ou un DB multi-instance. Vous pouvez également sélectionner des DB spécifiques dans une liste déroulante de DB disponibles.

2.11

Déconnexion temporaire d'appareils d'un réseau

Vous pouvez déconnecter des appareils individuels du sous-réseau. Comme la configuration de l'appareil n'est pas supprimée du projet, vous pouvez aisément restaurer la connexion à l'appareil.



Cliquez avec le bouton droit de la souris sur le port d'interface de l'appareil du réseau et sélectionnez la commande "Déconnecter du sous-réseau" dans le menu contextuel.

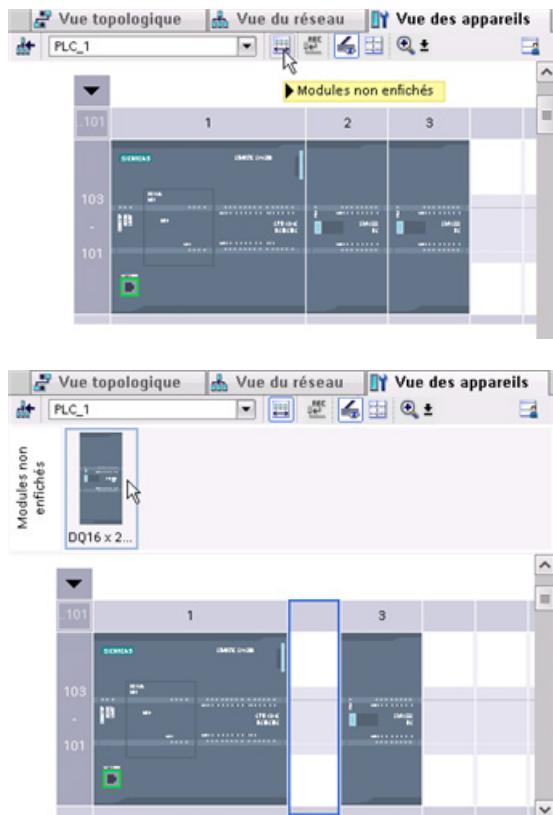
STEP 7 reconfigure les liaisons réseau, mais ne supprime pas l'appareil déconnecté du projet. Alors que la connexion réseau est supprimée, les adresses d'interface ne sont pas modifiées.



Vous devez faire passer la CPU à l'état ARRET lorsque vous chargez les nouvelles connexions réseau.

Pour reconnecter l'appareil, il vous suffit de créer une nouvelle connexion réseau au port de cet appareil.

2.12 "Retrait" virtuel facile des modules sans perte de configuration



STEP 7 propose une zone de stockage pour les modules "retirés". Vous avez la possibilité d'amener un module du châssis afin d'en enregistrer la configuration. Ces modules retirés sont enregistrés dans votre projet, ce qui permettra de les réinsérer ultérieurement sans avoir à redéfinir les paramètres de configuration.

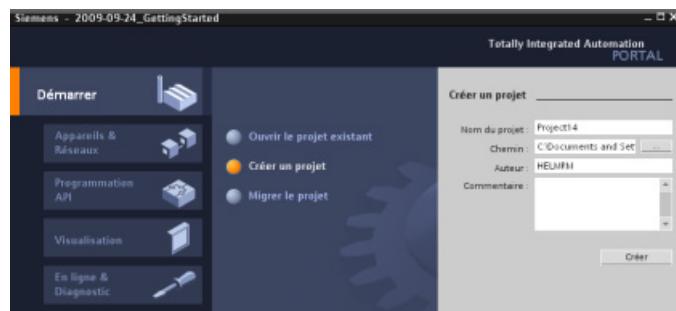
L'une des applications de cette fonctionnalité est la maintenance temporaire. Considérez l'hypothèse où vous auriez à attendre avant de pouvoir remplacer un module et où vous décideriez d'utiliser temporairement un module différent comme solution à court terme. Vous auriez la possibilité d'amener le module configuré du châssis dans les "Modules non enfoncés", puis d'insérer le module temporaire.

Mise en route

3.1

Créer un projet

Travailler avec STEP 7, c'est simple ! Voyez comme il est rapide de commencer à créer un projet.



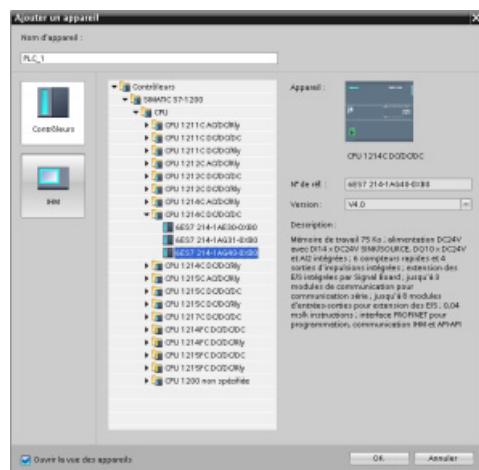
Dans le portail de démarrage, cliquez sur la tâche "Créer un nouveau projet".

Entrez un nom de projet et cliquez sur le bouton "Créer".



Après avoir créé le projet, sélectionnez le portail Appareils & Réseaux.

Cliquez sur la tâche "Ajouter un appareil".

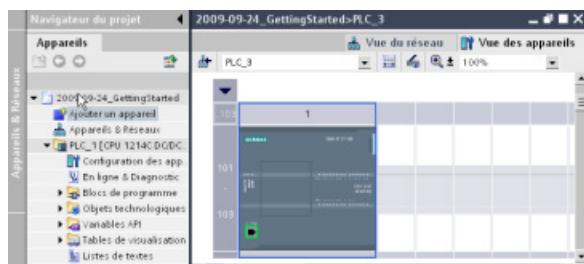


Sélectionnez la CPU à ajouter au projet :

1. Dans la boîte de dialogue "Ajouter un appareil", cliquez sur le bouton "SIMATIC PLC".
2. Sélectionnez une CPU dans la liste.
3. Pour ajouter la CPU sélectionnée au projet, cliquez sur le bouton "Ajouter".

Notez que l'option "Ouvrir vue des appareils" est sélectionnée. Lorsque vous cliquez sur "Ajouter" alors que cette option est sélectionnée, la "Configuration des appareils" de la vue du projet s'ouvre.

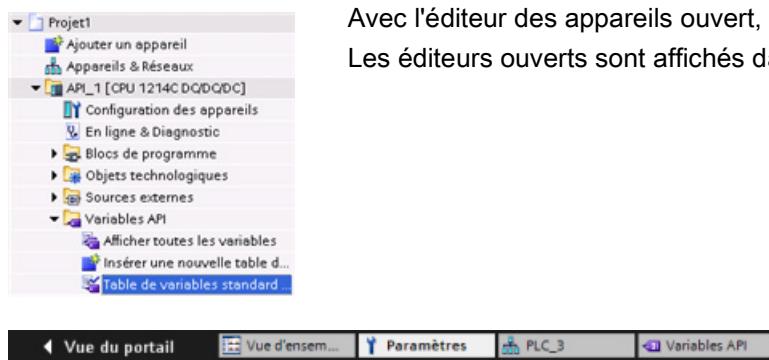
3.2 Créer des variables pour les E/S de la CPU



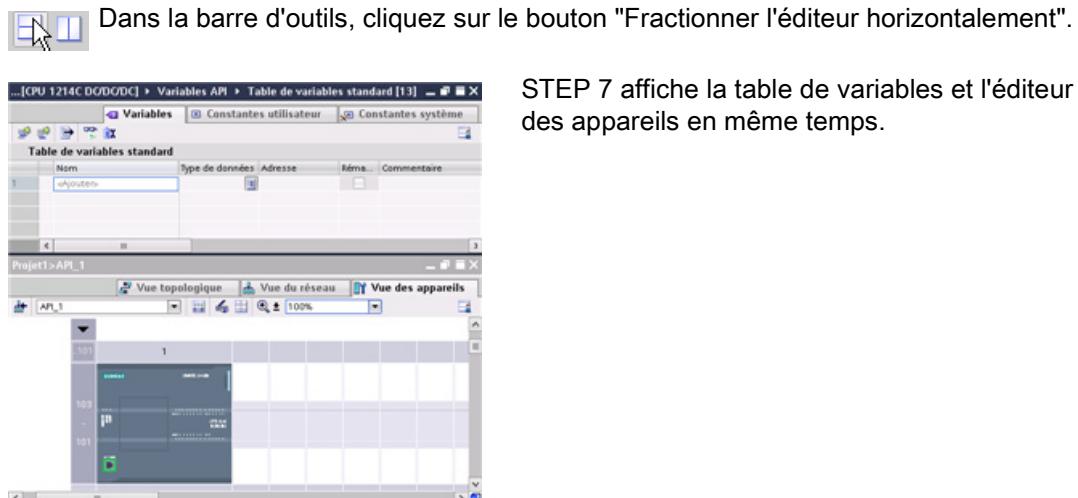
La vue des appareils affiche la CPU que vous avez ajouté.

3.2 Créer des variables pour les E/S de la CPU

Les "variables API" correspondent aux noms symboliques des E/S et des adresses. Après avoir créé une variable API, STEP 7 stocke la variable dans une table de variables. Tous les éditeurs dans votre projet (tels que l'éditeur de programmes, l'éditeur des appareils, l'éditeur de visualisation et l'éditeur de la table de visualisation) ont accès à cette table des variables.



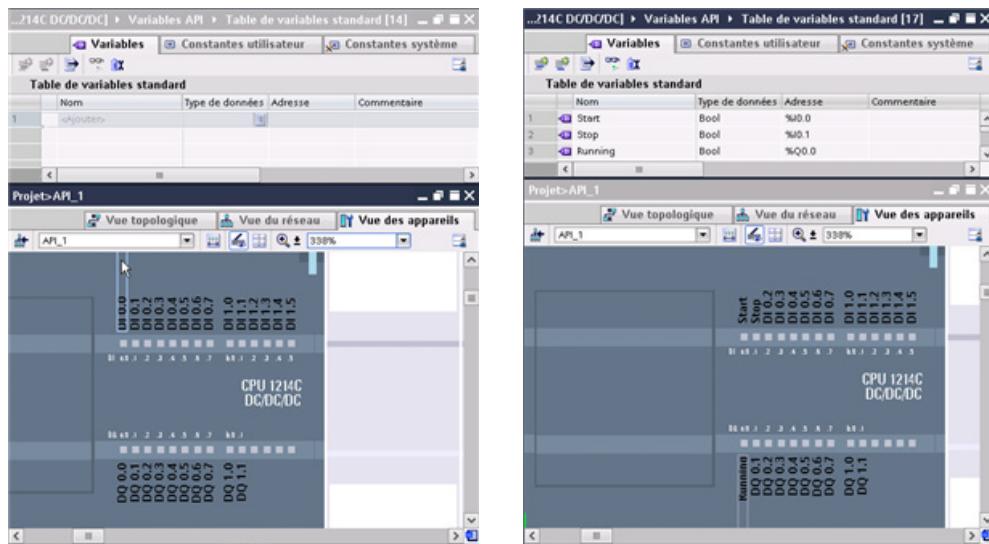
Avec l'éditeur des appareils ouvert, ouvrez la table de variables.
Les éditeurs ouverts sont affichés dans la barre d'édition.



STEP 7 affiche la table de variables et l'éditeur des appareils en même temps.

Agrandissez la configuration de l'appareil jusqu'à plus de 200% afin que les E/S de la CPU soit visibles et puissent être sélectionnées. Faites glisser les entrées et sorties à partir de la CPU vers la table des variables :

1. Sélectionnez I0.0 et amenez-la dans la première rangée de la table des variables.
2. Changez le nom de la variable de "I0.0" en "Start".
3. Amenez I0.1 dans la table des variables et changez son nom en "Stop".
4. Amenez Q0.0 (en bas de la CPU) dans la table des variables et changez son nom en "Running".

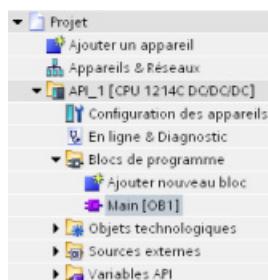


Avec les variables entrées dans la table des variables API, les variables sont disponibles pour votre programme utilisateur.

3.3

Créer un réseau simple dans votre programme utilisateur

Votre code de programme est composé d'opérations que la CPU exécute de façon séquentielle. Dans le présent exemple, utilisez les schémas à contacts (CONT) pour créer le code de programme. Le programme CONT correspond à une séquence de réseaux ressemblant aux barreaux d'une échelle.



Pour ouvrir l'éditeur de programmes, procédez comme suit :

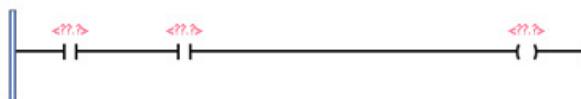
1. Dans l'arborescence du projet, naviguez jusqu'au bloc "Main [OB1]" dans le dossier "Blocs de programme".
2. Effectuez un double clic sur le bloc "Main [OB1]".

L'éditeur de programmes ouvre le bloc de programme (OB1).

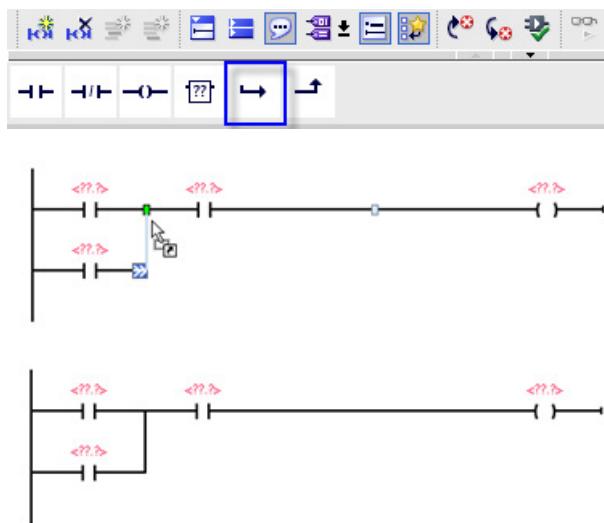
Utilisez les boutons sur les "Favoris" pour insérer les contacts et bobines sur le réseau.



1. Cliquez sur le bouton "Contact normalement ouvert" dans les "Favoris" pour ajouter un contact au réseau.
2. Pour le présent exemple, ajoutez un second contact.
3. Cliquez sur le bouton "Bobine de sortie" pour insérer une bobine.



Les "Favoris" présentent également un bouton pour créer une branche.

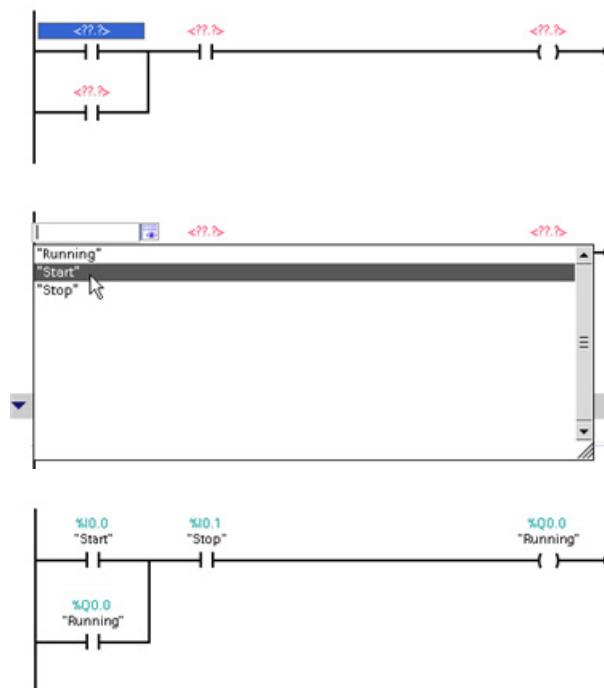


1. Sélectionner le rail gauche pour sélectionner la barre pour la branche.
2. Cliquez sur l'icône "Ouvrir branche" pour ajouter une branche à la barre du réseau.
3. Insérez un autre contact normalement ouvert dans la branche ouverte.
4. Faites glisser la flèche double sur un point de connexion (le carré vert sur le barreau) entre les deux contacts sur le premier barreau.

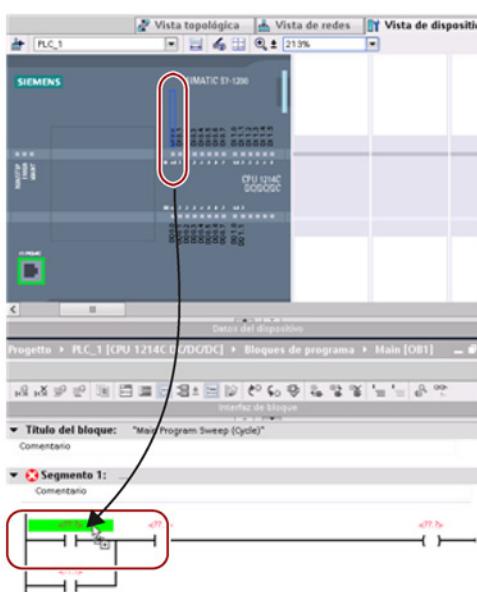
Pour enregistrer le projet, cliquez sur le bouton "Enregistrer projet" dans la barre d'outils. Notez que l'édition du barreau n'est pas terminée tant que vous n'avez pas effectuée d'enregistrement. Vous pouvez à présent affecter les noms de variables à ces instructions.

3.4 Utiliser les variables API de la table des variables pour l'adressages des instructions

En vous servant de la table des variables, vous pouvez rapidement entrer les variables API correspondant aux adresses des contacts et bobines.



1. Double-cliquez sur l'adresse par défaut <???.?> au-dessus du premier contact normalement ouvert.
2. Cliquez sur l'icône de sélection à droite de l'adresse, afin d'afficher les variables contenues dans la table des variables.
3. Dans la liste déroulante, sélectionnez "Start" pour le premier contact.
4. Répétez les étapes précédentes pour le second contact et sélectionnez la variable "Stop".
5. Pour la bobine et le contact d'auto-alimentation, sélectionnez la variable "Running".



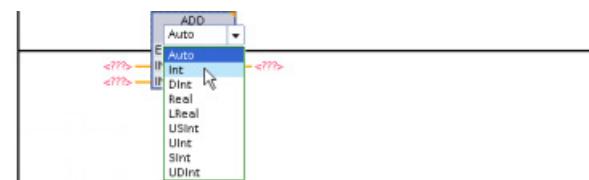
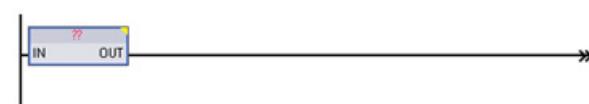
Vous pouvez également amener les adresses E/S directement à partir de la CPU. Fractionnez simplement la zone de travail de la vue du projet (Page 35).

Vous devez effectuer un agrandissement de la CPU d'au moins 200% pour pouvoir sélectionner les E/S.

Vous pouvez amener les E/S de la "Configuration des appareils" dans la CPU sur l'instruction CONT dans l'éditeur de programmes, afin de créer non seulement l'adresse de l'instruction, mais également une entrée dans la table des variables API.

3.5 Ajouter une "boîte" d'opérations

L'éditeur de programmes contient une "boîte" d'opérations générique. Après avoir inséré cette boîte d'opérations, vous sélectionnez le type d'opération, p. ex. une opération ADD, dans une liste déroulante.

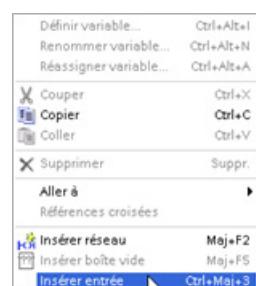


Cliquez sur la "boîte" d'opérations générique dans la barre d'outils "Favoris".

La "boîte" d'opérations générique propose de nombreuses opérations. Dans notre exemple, créez une opération ADD :

1. Cliquez sur l'angle jaune de la boîte d'opérations pour afficher la liste déroulante d'opérations.
2. Faites défiler la liste et sélectionnez l'opération ADD.
3. Cliquez sur l'angle jaune de "?" pour sélectionner le type de données des entrées et sorties.

Vous pouvez à présent entrer les variables (ou adresses mémoire) pour les valeurs à utiliser avec l'opération ADD.



Pour certaines opérations, vous pouvez également créer des entrées supplémentaires :

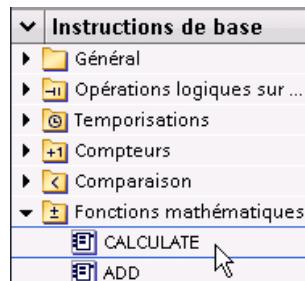
1. Cliquez sur l'une des entrées à l'intérieur de la boîte.
2. Effectuez un clic droit de la souris pour afficher le menu contextuel, puis sélectionnez la commande "Insérer entrée".



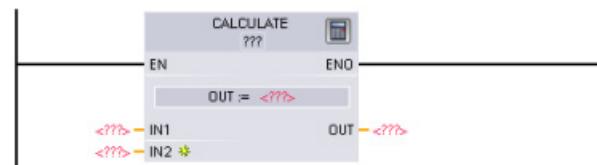
L'opération ADD dispose à présent de trois entrées.

3.6 Utilisez l'opération CALCULER pour une équation mathématique complexe

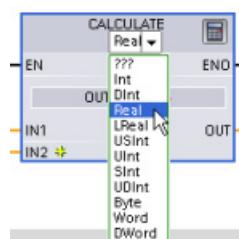
L'instruction Calculate (Page 117) vous permet de créer une fonction mathématique qui fonctionne sur des paramètres à entrées multiples pour produire le résultat, selon l'équation que vous définissez.



Dans l'arborescence d'opération Basic, agrandissez le fichier de fonctions Mathématiques. Double-cliquez sur l'opération Calculate pour insérer l'opération dans votre programme utilisateur.



L'opération Calculate non-configuration fournit deux paramètres d'entrée et un paramètre de sortie.

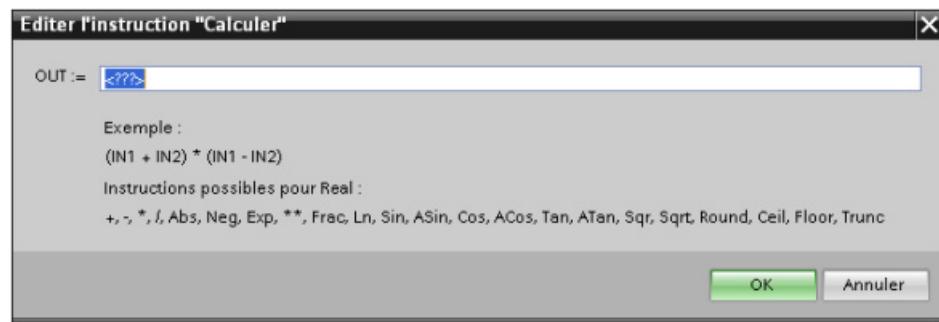


Cliquez sur "???" et sélectionnez les types de données pour les paramètres d'entrée et de sortie. (Les paramètres d'entrée et de sortie doivent tous être du même type de données.)

Pour cet exemple, sélectionnez le type de données "Real".



Cliquez sur l'icône "Éditer équation" pour saisir l'équation.



3.6 Utilisez l'opération CALCULER pour une équation mathématique complexe

Pour cet exemple, saisissez l'équation suivante pour établir une valeur brute analogique. (Les désignations "Entrée" et "Sortie" correspondent aux paramètres de l'opération Calculer.)

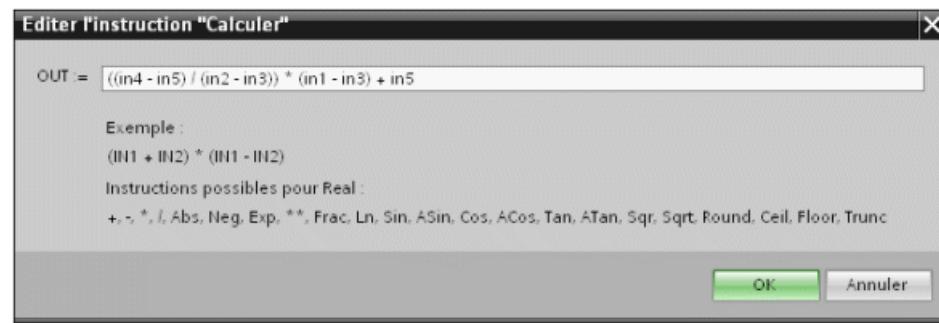
$$\text{Out value} = ((\text{Out high} - \text{Out low}) / (\text{In high} - \text{In low})) * (\text{In value} - \text{In low}) + \text{Out low}$$

$$\text{Out} = ((\text{in4} - \text{in5}) / (\text{in2} - \text{in3})) * (\text{in1} - \text{in3}) + \text{in5}$$

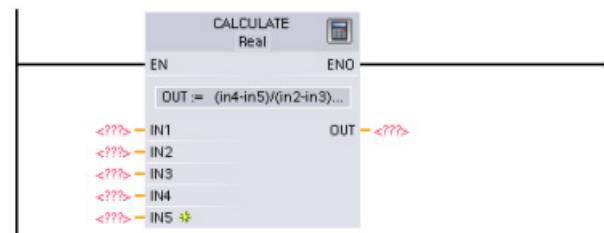
Si :	Out value	(Out)	Valeur de sortie graduée
	In value	(in1)	Valeur d'entrée analogique
	In high	(in2)	Limite supérieure pour la valeur d'entrée graduée
	In low	(in3)	Limite inférieure pour la valeur d'entrée graduée
	Out high	(in4)	Limite supérieure pour la valeur de sortie graduée
	Out low	(in5)	Limite inférieure pour la valeur de sortie normalisée

Dans le champ "Éditer Calculer", saisissez l'équation avec les noms de paramètre :

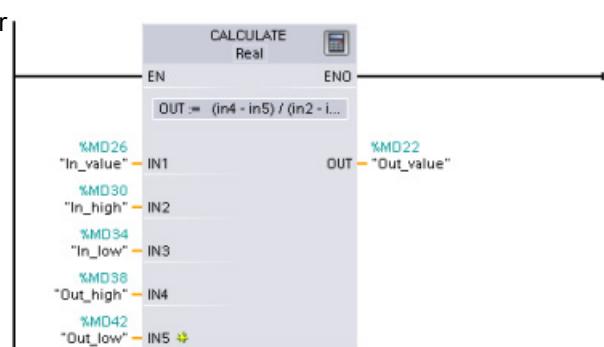
$$\text{OUT} = ((\text{in4} - \text{in5}) / (\text{in2} - \text{in3})) * (\text{in1} - \text{in3}) + \text{in5}$$



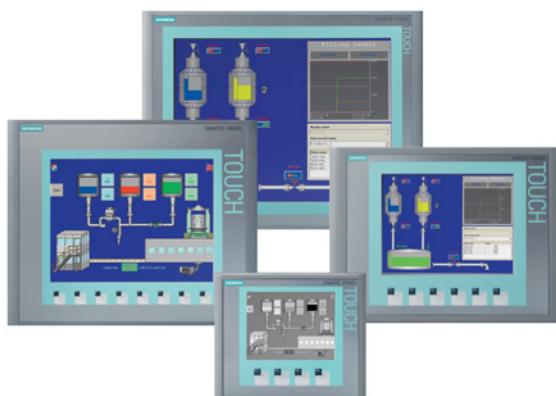
Lorsque vous cliquez sur "OK", l'instruction Calculate crée les entrées nécessaires à l'opération.



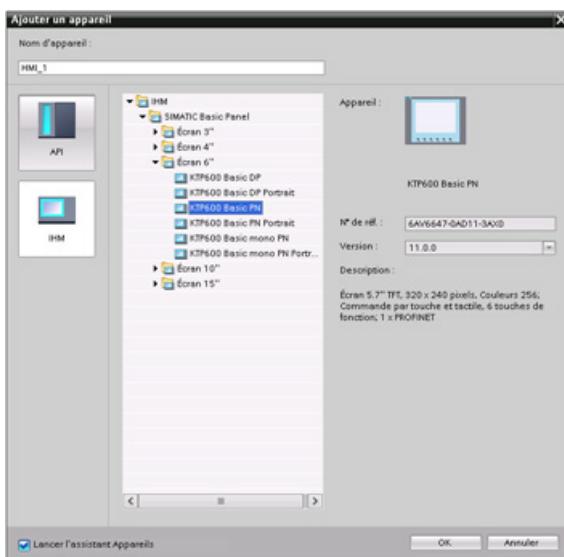
Saisissez les noms de variables pour les valeurs qui correspondent aux paramètres.



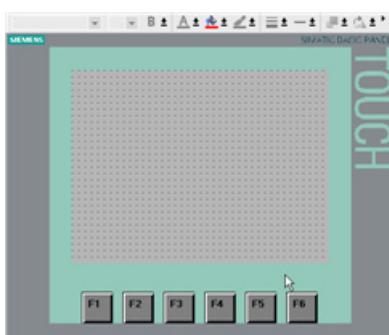
3.7 Ajouter un appareil IHM au projet



Ajouter un appareil IHM à votre projet est facile !



1. Double-cliquez sur l'icône "Ajouter un appareil".
 2. Cliquez sur le bouton "SIMATIC HMI" dans la boîte de dialogue "Ajouter un appareil".
 3. Sélectionnez l'appareil IHM souhaité dans la liste.
- Vous pouvez lancer l'assistant IHM qui vous aidera à configurer les écrans de l'appareil IHM.
4. Cliquez sur "OK" pour ajouter l'appareil IHM à votre projet.



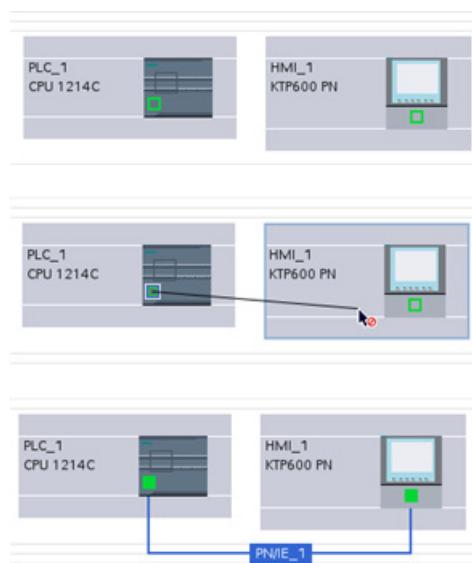
Le portail TIA ajoute l'appareil IHM au projet.

Le portail TIA propose un assistant IHM qui vous aide à configurer tous les écrans et toutes les structures pour votre appareil IHM.

Si vous n'exécutez pas l'assistant IHM, le portail TIA crée un écran IHM simple par défaut. Vous pouvez ajouter ultérieurement des écrans ou des objets supplémentaires sur des écrans.

3.8

Créer une liaison réseau entre la CPU et l'appareil IHM

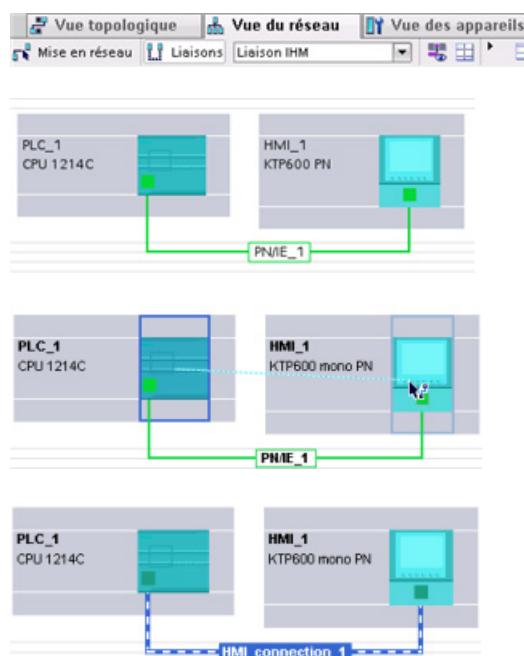


Créer un réseau est facile !

- Dans "Appareils & réseaux", sélectionnez la vue du réseau pour afficher la CPU et l'appareil IHM.
- Pour créer un réseau PROFINET, tracez une ligne entre le carré vert (port Ethernet) de l'un des appareils et le carré vert de l'autre appareil. Une liaison réseau est créée pour les deux appareils.

3.9

Créer une liaison IHM pour partager des variables



En créant une liaison IHM entre deux appareils, vous pouvez facilement échanger des variables entre ces deux appareils.

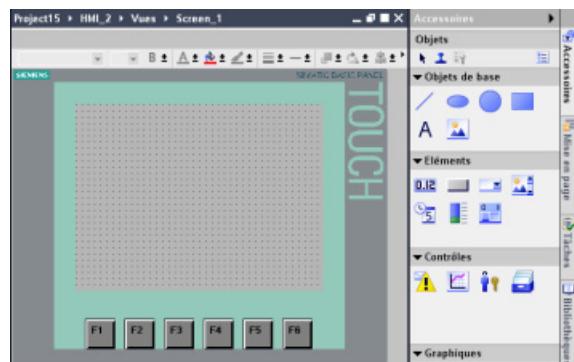
- La connexion réseau étant sélectionnée, cliquez sur le bouton "Connexions" et sélectionnez dans la liste déroulante "Connexion IHM".
- Lorsque la liaison IHM est établie, les deux appareils s'affichent en bleu.
- Sélectionnez la CPU et tracez la ligne vers l'appareil IHM.
- La liaison IHM vous permet de configurer des variables IHM en sélectionnant une liste de variables API.

Vous pouvez utiliser d'autres méthodes pour créer une liaison IHM :

- Faire glisser une variable API de la table des variables API, de l'éditeur de programme ou de l'éditeur de configuration des appareils dans l'éditeur d'écran IHM créé automatiquement une liaison IHM.
- Utiliser l'assistant IHM pour rechercher l'API créée automatiquement la liaison IHM.

3.10 Créer un écran IHM

Même si vous n'utilisez pas l'assistant IHM, la configuration d'un écran IHM est facile.



STEP 7 fournit un ensemble de bibliothèques standard pour insérer des formes basiques, des éléments interactifs et même des graphiques standards.

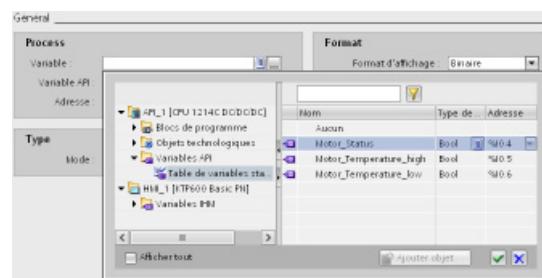
Pour ajouter un élément, amenez-le simplement dans l'écran par glisser-déplacer. Allez dans les propriétés de l'élément (dans la fenêtre d'inspection) pour configurer son apparence et son comportement.



Vous pouvez également créer des éléments sur votre écran en faisant glisser des variables API de l'arborescence de projet ou de l'éditeur de programme dans l'écran IHM. La variable API devient un élément de l'écran. Vous pouvez alors vous servir des propriétés pour modifier les paramètres de cet élément.

3.11 Sélectionner une variable API pour l'élément IHM

Après avoir créé l'élément sur votre écran, utilisez les propriétés de l'élément pour affecter une variable API à l'élément. Cliquez sur le bouton de sélection du champ de la variable afin d'afficher les variables API de la CPU.



Vous pouvez également faire glisser des variables API de l'arborescence de projet dans l'écran IHM. Affichez les variables API dans la vue "Détails" de l'arborescence de projet et faites glisser la variable désirée dans l'écran IHM.

Simplification des concepts API

4.1

Tâches réalisées à chaque cycle

Chaque cycle comprend l'écriture dans les sorties, la lecture des entrées, l'exécution des instructions du programme utilisateur et la maintenance système ou le traitement d'arrière-plan.

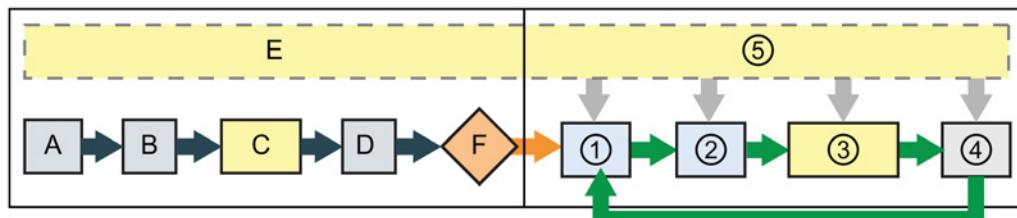


On parle parfois de cycle d'exploration. Dans les conditions normales, toutes les E/S TOR et analogiques sont actualisées en synchronisme avec le cycle à l'aide d'une zone de mémoire interne appelée mémoire image du processus. La mémoire image contient un cliché instantané des entrées et sorties physiques de la CPU, du Signal Board et des modules d'entrées-sorties.

- La CPU lit les entrées physiques juste avant l'exécution du programme utilisateur et stocke les valeurs d'entrée dans la mémoire image des entrées. Cela garantit que ces valeurs restent cohérentes pendant toute l'exécution des instructions utilisateur.
- La CPU exécute la logique des instructions utilisateur et actualise les valeurs des sorties dans la mémoire image des sorties sans les écrire dans les sorties physiques réelles.
- Une fois le programme utilisateur exécuté, la CPU écrit les sorties résultantes de la mémoire image des sorties dans les sorties physiques.

4.1 Tâches réalisées à chaque cycle

Ce procédé assure une logique cohérente durant l'exécution des instructions utilisateur pour un cycle donné et empêche le papillotement des sorties physiques dont l'état peut changer à de nombreuses reprises dans la mémoire image des sorties.



MISE EN ROUTE

- A Efface la zone de mémoire I (image)
- B Met la zone de mémoire (image) des sorties Q à zéro, à la dernière valeur, ou à la valeur de remplacement selon la configuration et met à zéro PB, PN et les sorties AS-i
- C Met la mémoire non rémanente M et les blocs de données à leur valeur initiale et autorise une alarme et une durée cyclique pour les événements quotidiens.
Exécute les OB de démarrage.
- D Copie l'état des entrées physiques dans la mémoire image des entrées.
- E Enregistre tous les événements d'alarme dans la file d'attente en vue de leur traitement après être passé à l'état MARCHÉ
- F Valide l'écriture de la mémoire Q dans les sorties physiques.

MARCHÉ

- ① Ecrit la mémoire image des sorties dans les sorties physiques.
- ② Copie l'état des entrées physiques dans la mémoire image des entrées.
- ③ Exécute les OB de cycle du programme.
- ④ Réalise des test d'auto-diagnostic.
- ⑤ Traite les alarmes et la communication à n'importe quel moment du cycle.

Vous pouvez modifier le comportement par défaut d'un module en y désactivant cette actualisation automatique des E/S. Vous pouvez également lire et écrire directement les valeurs d'E/S TOR et analogiques dans les modules lors de l'exécution d'une instruction. La lecture directe d'entrées physiques n'entraîne pas de mise à jour de la mémoire image des entrées. L'écriture directe dans des sorties physiques provoque l'actualisation de la mémoire image des sorties et de la sortie physique concernée.

4.2

Etats de fonctionnement de la CPU

La CPU a trois états de fonctionnement : l'état ARRET (STOP), l'état MISE EN ROUTE (STARTUP) et l'état MARCHE (RUN). Des DEL d'état en face avant de la CPU signalent l'état de fonctionnement en cours.

- A l'état ARRET, la CPU n'exécute pas le programme et vous pouvez y charger un projet. La DEL MARCHE/ARRET est jaune fixe.
- A l'état MISE EN ROUTE, la CPU exécute la logique de démarrage (en présence d'une telle logique). Les événements d'alarme ne sont pas traités par la CPU pendant l'état mise en route. La DEL MARCHE/ARRET clignote alternativement du vert au jaune.
- A l'état MARCHE, le cycle est exécuté de manière répétée. Des événements d'alarme peuvent survenir et être traités par la CPU à tout moment durant la phase de cycle du programme. Vous pouvez télécharger certaines parties du projet à l'état MARCHE. La DEL MARCHE/ARRET est verte fixe.

La CPU accepte un démarrage à chaud pour le passage à l'état MARCHE. Un démarrage à chaud ne comporte pas de réinitialisation mémoire, mais vous pouvez ordonner une réinitialisation de mémoire à partir de STEP 7. Une réinitialisation de mémoire efface toute la mémoire de travail, efface les zones de mémoire rémanentes ou non-rémanentes, copie la mémoire de chargement sur la mémoire de travail, et définit des sorties pour la "Réaction à l'ARRET DE CPU" configurée. Un effacement général n'efface pas la mémoire tampon de diagnostic ni l'adresse IP sauvegardée de manière permanente. Un démarrage à chaud initialise toutes les données système et utilisateur non rémanentes.

Vous pouvez configurer le paramètre "le démarrage après la MISE SOUS TENSION" de la CPU est achevé avec la méthode de redémarrage à l'aide de STEP 7. Cet élément de configuration apparaît sous la Configuration de l'Appareil pour la CPU sous la rubrique Démarrage. A la mise sous tension, la CPU exécute une séquence de vérifications du diagnostic de mise sous tension et effectue l'initialisation du système. Pendant l'initialisation du système, la CPU efface tous les mémentos non rémanents et redonne à tous les DB non rémanents leurs valeurs initiales. La CPU passe alors à l'état de mise en route approprié. Certaines erreurs empêcheront le passage de la CPU à l'état MARCHE. La CPU accepte les états de mise en route suivants : état ARRET, "Passage à l'état MARCHE après un démarrage à chaud" et "Passage à l'état précédent après un démarrage à chaud".

IMPORTANT

Configuration mode de redémarrage à chaud

La CPU peut passer à l'état ARRET en cas de défaillances réparables, telles que la défaillance d'un module d'entrées-sorties remplaçable, ou en cas de défaillances temporaires, telles qu'une perturbation de la ligne d'alimentation ou une mauvaise mise en route.

Si la CPU a été configurée avec "Démarrage à chaude - mode de fonctionnement avant la mise hors tension", elle ne repassera pas à l'état MARCHE une fois la défaillance réparée ou résolue tant qu'elle ne reçoit pas une nouvelle commande de STEP 7 lui demandant de passer à l'état MARCHE. Si la CPU ne reçoit pas de nouvelle commande, l'état ARRET est maintenu comme état précédent la mise hors tension.

Les CPU conçues pour fonctionner indépendamment d'une connexion STEP 7 doivent typiquement être configurées avec "Démarrage à chaud - MARCHE" de manière à ce que la CPU puisse repasser à l'état MARCHE avec une mise sous tension après la résolution de l'état de défaillance.



La CPU ne dispose pas de commutateur physique pour réaliser le changement d'état de fonctionnement. Pour modifier l'état de fonctionnement de la CPU, vous disposez dans STEP 7 des outils suivants :

- Boutons "Arrêt" et "Marche" dans la barre d'outils STEP 7
- Panneau de commande CPU dans les outils en ligne

Vous pouvez aussi insérer une instruction STP dans votre programme pour faire passer la CPU à l'état ARRET. Cela vous permet d'arrêter l'exécution de votre programme selon la logique. Le serveur Web (Page 264) possède également une page permettant de réaliser le changement d'état de fonctionnement.

4.3

Exécution du programme utilisateur

La CPU fournit les types suivants de blocs de code qui vous permettent de créer une structure efficace pour votre programme utilisateur :

- Les blocs d'organisation (OB) définissent la structure du programme. Certains OB ont un comportement et des événements de démarrage prédéfinis, mais vous pouvez également créer des OB avec des événements de démarrage personnalisés (Page 58).
- Les fonctions (FC) et blocs fonctionnels (FB) contiennent le code de programme qui correspond à des tâches ou des combinaisons de paramètres spécifiques. Chaque FC ou FB fournit un jeu de paramètres d'entrée et de sortie pour partager les données avec le bloc appelant. Un FB utilise également un bloc de données associé (appelé DB d'instance) pour conserver l'état de valeurs entre les exécutions, utilisable par d'autres blocs dans le programme.
- Les blocs de données (DB) mémorisent des données qui peuvent être utilisées par les blocs de programme.

La dimension du programme utilisateur, les données et la configuration sont limitées par la disponibilité en mémoire de chargement et mémoire de travail dans la CPU (Page 15). Il n'y a pas de limite spécifique au nombre de chaque bloc OB, FC, FB et DB individuel. Cependant, le nombre total de blocs est limité à 1 024.

4.3.1

Traitements du cycle à l'état MARCHE

A chaque cycle, la CPU écrit dans les sorties, lit les entrées, exécute le programme utilisateur, actualise les modules de communication et réagit aux événements d'alarme utilisateur et aux demandes de communication. Les demandes de communication sont traitées périodiquement pendant tout le cycle.

Ces actions (à l'exception des événements d'alarme utilisateur) sont assurées régulièrement et à la suite les unes des autres. Les événements d'alarme utilisateur activés sont gérés selon leur priorité dans l'ordre où ils se produisent. Pour les événements d'alarme, la CPU lit les entrées, exécute l'OB, puis écrit dans les sorties en utilisant la mémoire image partielle (MIP) associée, le cas échéant.

Le système garantit que le cycle s'achèvera au bout d'une durée appelée temps de cycle maximum. Dans le cas contraire, un événement d'erreur de temps est généré.

- Chaque cycle commence par l'extraction des valeurs en cours des sorties TOR et analogiques de la mémoire image et leur écriture dans les sorties physiques de la CPU, du SB et des SM configurées pour l'actualisation automatique des E/S (configuration par défaut). Lorsqu'on accède à une sortie physique par une instruction, la mémoire image des sorties et la sortie physique elle-même sont toutes deux actualisées.
- Chaque cycle se poursuit par la lecture des valeurs en cours des entrées TOR et analogiques dans la CPU, le SB et les SM configurées pour l'actualisation automatique des E/S (configuration par défaut), puis par l'écriture de ces valeurs dans la mémoire image des entrées. Lorsqu'on accède à une entrée physique par une instruction, l'instruction accède à la valeur de l'entrée physique mais la mémoire image des entrées n'est pas actualisée.

- Une fois les entrées lues, le programme utilisateur est exécuté de la première instruction jusqu'à l'instruction finale. Cela comprend tous les OB du cycle de programme avec leurs FC et FB associés. Les OB du cycle de programme sont exécutés par ordre de numéro d'OB, l'OB de plus petit numéro étant exécuté en premier.

Le traitement de la communication se produit périodiquement pendant tout le cycle, interrompant éventuellement l'exécution du programme utilisateur.

Les contrôles d'auto-diagnostic comprennent des contrôles périodiques du système ainsi que des contrôles de l'état des modules d'E/S.

Des alarmes peuvent se produire à tout endroit du cycle ; elles sont déclenchées sur événement. Lorsqu'un événement se produit, la CPU interrompt le cycle et appelle l'OB qui a été configuré pour traiter cet événement. Lorsque l'OB a fini de traiter l'événement, la CPU reprend l'exécution du programme utilisateur à l'endroit où elle s'était interrompue.

4.3.2 Les OB vous aident à structurer votre programme utilisateur

Les OB pilotent l'exécution du programme utilisateur. Des événements spécifiques dans la CPU déclenchent l'exécution d'un bloc d'organisation. Les OB ne peuvent pas s'appeler les uns les autres ni être appelés dans une FC ou un FB. Seul un événement tel qu'une alarme de diagnostic ou un intervalle de temps peut lancer l'exécution d'un OB. La CPU traite les OB en fonction de leur classe de priorité respective, les OB de priorité supérieure s'exécutant avant les OB de priorité inférieure. La classe de priorité la plus faible est 1 (pour le cycle du programme principal) et la classe de priorité la plus élevée est 26.

4.3.3 Priorités d'exécution et mise en file d'attente des événements

Le traitement de la CPU est commandé par des événements. Un événement déclenche un OB d'alarme qu'il faut exécuter. Vous pouvez définir l'OB d'alarme pour un événement lors de la création du bloc, lors de la configuration de l'appareil ou à l'aide d'une opération ATTACH ou DETACH. Certains événements se produisent sur une base régulière, tels les événements cycle de programme ou les événements alarme cyclique. D'autres événements ne se produisent qu'une fois, tels l'événement démarrage et les événements alarme temporisée. Certains événements se produisent lorsque le matériel déclenche un événement, par exemple un événement front sur une entrée ou un événement compteur rapide. Des événements tels que l'erreur de diagnostic et l'erreur de temps ne se produisent que lorsqu'une erreur se produit. Les priorités et files d'attente d'événements sont utilisées pour déterminer l'ordre de traitement des OB d'alarme d'événement.

La CPU traite des événements par ordre de priorité dans lequel 1 est la priorité la plus faible et 26 est la priorité la plus élevée. Avant le V4.0 de la CPU S7-1200, chaque type d'OB appartenait à une classe de priorité fixe (1 à 26). A partir de la version V4.0, vous pouvez affecter une classe de priorité à chaque OB que vous configurez. Vous configurez le nombre de priorités dans les attributs des propriétés OB.

Etats d'exécution interruptible et non-interruptible

Les OB (Page 58) s'exécutent dans l'ordre de priorité des événements qui les déclenchent. A partir de la version V4.0, vous pouvez configurer l'exécution de l'OB pour qu'elle soit interruptible ou non interruptible. Notez que les OB de ce cycle de programme sont toujours interruptibles, mais vous pouvez configurer tous les autres OB pour qu'ils soient interruptibles ou non interruptibles.

Si vous définissez l'état interruptible, alors si un OB s'exécute et qu'un événement de priorité plus haute se produit avant que l'OB achève son exécution, l'OB en cours s'interrompt pour permettre à l'OB d'un événement ayant la priorité la plus élevée de s'exécuter. L'événement ayant la priorité la plus élevée s'exécute, et à son achèvement, l'OB qui a été interrompu reprend. Lorsque des événements multiples se produisent alors qu'un OB alarme s'exécute, la CPU traite ces événements dans l'ordre de priorité.

Si vous ne définissez pas un état interruptible, alors un OB s'exécute jusqu'à son achèvement lorsqu'il est déclenché indépendamment de tout autre événement qui se déclenche pendant sa durée de fonctionnement.

Envisagez les deux cas suivants où des événements d'alarme déclenchent un OB cyclique et un OB temporisé. Dans les deux cas, l'OB temporisé (OB201) n'a pas d'affectation de partition de mémoire image et s'exécute à la priorité 4. L'OB cyclique (OB200) a une affectation de partition de mémoire image de PIP1 et s'exécute à la priorité 2. Les illustrations suivantes montrent la différence d'exécution entre des états d'exécution interruptible et non interruptible :

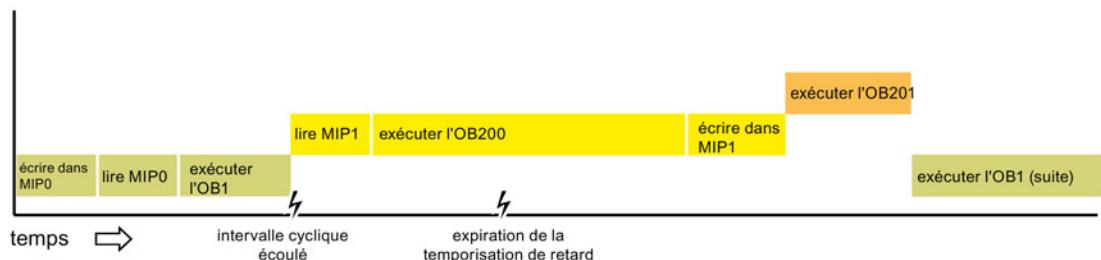


Figure 4-1 Cas 1 : Exécution d'OB non interruptible

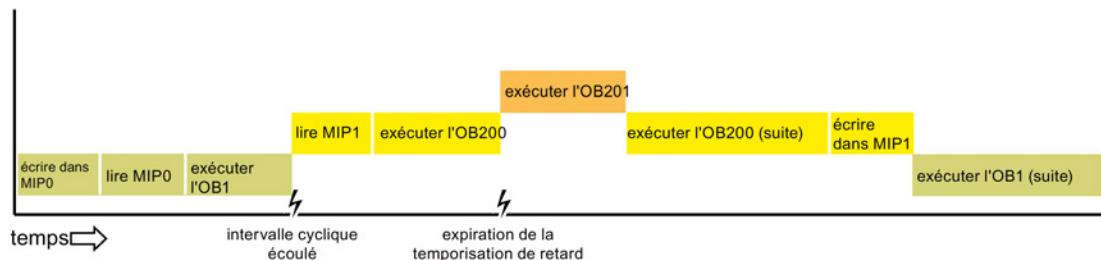


Figure 4-2 Cas 2 : exécution d'OB interruptible

Remarque

Si vous configurez l'état d'exécution d'OB pour être non interruptible, alors un OB d'erreur de temps ne peut pas interrompre d'autres OB que des OB de cycle de programme. Avant la V4.0 de la CPU du S7-1200, un OB d'erreur de temps pouvait interrompre n'importe quel OB d'exécution. A partir de la V4.0, vous devez configurer l'exécution d'un OB pour être interruptible si vous voulez qu'un OB d'erreur de temps (ou n'importe quel autre OB d'une priorité plus élevée) soit capable d'interrompre l'exécution des OB qui ne sont pas des OB de cycle de programme.

Comprendre les priorités d'exécution et la mise en file d'attente des événements

La CPU limite le nombre d'événements en attente provenant d'une source unique, avec une file d'attente différente pour chaque type d'événement. Lorsque la limite d'événements en attente pour un type d'événement donné est atteinte, l'événement suivant qui survient est perdu. Vous pouvez utiliser une OB d'alarme d'erreur de temps pour répondre à des débordements de file d'attente.

Chaque événement de CPU a une priorité associée. En général, les événements sont traités par la CPU selon l'ordre de priorité (priorité la plus élevée en premier). Les événements de même priorité sont traités selon le principe "premier arrivé, premier servi".

Tableau 4- 1 Événements d'OB

Événement	Quantité autorisée	Priorité d'OB par défaut
Cycle de programme	1 événement de cycle de programme Plusieurs OB autorisés	1 ⁴
Mise en route	1 événement de démarrage ¹ Plusieurs OB autorisés	1 ⁴
Alarme temporisée	Jusqu'à 4 événements horaires 1 OB par événement	3
Alarme cyclique	Jusqu'à 4 événements 1 OB par événement	8
Alarme de processus	Jusqu'à 50 événements d'alarme de processus ² 1 OB par événement, mais vous pouvez utiliser le même OB pour des événements multiples	18 18
Erreur de temps	1 événement (seulement s'il est configuré) ³	22 ou 26 ⁴
Erreur de diagnostic	1 événement (seulement s'il est configuré)	5
Débrochage ou enfonchage de modules	1 événement	6
Défaillance du châssis ou de la station	1 événement	6
Heure	Jusqu'à 2 événements	2
Etat	1 événement	4

Événement	Quantité autorisée	Priorité d'OB par défaut
Mettre à jour	1 événement	4
Profil	1 événement	4

- ¹ L'événement démarrage et l'événement cycle de programme ne se produisent jamais en même temps, car l'événement démarrage s'exécute jusqu'à son achèvement avant que l'événement cycle de programme ne commence.
- ² Vous pouvez avoir plus de 50 OB d'événements alarme de processus si vous utilisez les instructions DETACH et ATTACH.
- ³ Vous pouvez configurer la CPU afin qu'elle reste à l'état MARCHE si le temps de cycle de balayage dépasse le temps de balayage maximal ou vous pouvez utiliser l'instruction RE_TRIGR pour réinitialiser le temps de cycle. Toutefois, la CPU passera à l'état ARRET la deuxième fois que le temps de cycle dépasse le temps de cycle maximal.
- ⁴ La priorité pour une nouvelle CPU V4.0 ou V4.1 est fixée à 22. Si vous remplacez une CPU V3.0 par une CPU V4.0 ou V4.1, la priorité est 26 : la priorité en vigueur pour la V3.0. Dans tous les cas, le champ de priorité peut être édité et vous pouvez fixer la priorité à une valeur comprise entre 22 et 26.

Pour plus de détails, reportez-vous à la rubrique "Remplacer une CPU V3.0 par une CPU V4.1 (Page 453)".

En outre, la CPU reconnaît d'autres événements qui n'ont pas d'OB associés. La table suivante décrit ces événements et les actions de la CPU correspondantes :

Tableau 4- 2 événements supplémentaires

Événement	Description	action CPU
Erreur d'accès aux E/S	Erreur lecture/écriture E/S directe	La CPU enregistre la première occurrence dans le tampon de diagnostic et reste à l'état MARCHE.
Erreur de temps de cycle max.	La CPU dépasse le double du temps de cycle configuré	La CPU enregistre la première occurrence dans le tampon de diagnostic et reste à l'état MARCHE.
Erreur d'accès à la périphérie	Erreur d'E/S pendant la mise à jour de la mémoire image	La CPU enregistre la première occurrence dans le tampon de diagnostic et reste à l'état MARCHE.
Erreur de programmation	Erreurs d'exécution de programme	Si le bloc comportant l'erreur fournit une gestion d'erreur, il met à jour la structure d'erreur ; sinon, la CPU enregistre l'erreur dans le tampon de diagnostic et reste à l'état MARCHE.

Temps d'attente

Le temps d'attente d'un événement d'alarme (le temps qui s'écoule entre la notification à la CPU qu'un événement s'est produit et le moment où la CPU exécute la première instruction dans l'OB gérant cet événement) est d'environ 175 µs, à condition qu'un OB de cycle de programme soit le seul programme de service d'événement actif au moment de l'événement d'alarme.

4.4

Zones de mémoire, adressage et types de données

La CPU dispose des zones de mémoire suivantes pour stocker le programme utilisateur, les données et la configuration :

- La mémoire de chargement est une mémoire rémanente pour le programme utilisateur, les données et la configuration. Lorsqu'un projet est chargé dans la CPU, il est d'abord stocké dans la mémoire de chargement. Cette zone se trouve soit sur une carte mémoire (en présence d'une telle carte), soit dans la CPU. Cette zone de mémoire rémanente est conservée en cas de coupure de courant. Vous pouvez augmenter la quantité de mémoire de chargement pour les enregistrements de données en installant une carte mémoire.
- La mémoire de travail est une mémoire volatile pour certains éléments du projet utilisateur pendant l'exécution du programme utilisateur. La CPU copie certains éléments du projet de la mémoire de chargement dans la mémoire de travail. Cette zone de mémoire volatile est perdue en cas de coupure de courant et est restaurée par la CPU au retour de la tension.
- La mémoire rémanente constitue un stockage permanent pour une quantité limitée de valeurs de mémoire de travail. La zone de mémoire rémanente sert à stocker les valeurs d'adresses de mémoire utilisateur choisies pendant une coupure de courant. Si une mise hors tension ou une coupure de courant se produit, la CPU restaurera ces valeurs rémanentes à la mise sous tension suivante.



Une carte mémoire SIMATIC optionnelle s'utilise comme mémoire supplémentaire pour stocker votre programme utilisateur ou comme moyen de transfert de votre programme. Si vous utilisez la carte mémoire, la CPU exécute le programme qui est dans la carte mémoire et non celui qui est dans la mémoire de la CPU.

Vérifiez que la carte mémoire n'est pas protégée en écriture. Faites glisser le commutateur de protection pour l'éloigner de la position "verrou".

Utilisez la carte mémoire SIMATIC optionnelle comme carte de programme, comme carte de transfert, pour collecter des fichiers journaux, ou pour effectuer une mise à jour du firmware.

- Utilisez la carte transfert pour copier votre projet vers de multiples CPU sans utiliser STEP 7. La carte transfert copie un projet stocké sur la carte vers la mémoire de la CPU. Vous devez retirer la carte transfert après avoir copié le programme dans la CPU.
- La carte programme remplace la mémoire CPU ; toutes les fonctions CPU sont pilotées par la carte programme. L'insertion d'une carte programme efface toute la mémoire de chargement interne de la CPU (notamment le programme utilisateur et les E/S forcées en permanence). La CPU exécute alors le programme utilisateur dans la carte programme.
- Vous pouvez également utiliser la carte programme pour collecter les fichiers d'enregistrement de données (Page 127). La carte programme fournit plus de mémoire que la mémoire interne de la CPU. La fonction serveur Web (Page 263) de la CPU vous permet de télécharger les fichiers d'enregistrement des données vers un ordinateur.
- Vous pouvez également utiliser une carte mémoire pour effectuer une mise à jour du firmware. Pour obtenir des instructions, référez-vous au *Manuel système de l'automate programmable S7-1200*.

Remarque

La carte programme **doit** rester dans la CPU. Si vous retirez la carte programme, la CPU passe à l'état ARRET.

4.4.1 Types de données pris en charge par le S7-1200

Les types de données servent à indiquer à la fois la taille d'un élément de données et la façon dont les données doivent être interprétées. Chaque paramètre d'une instruction accepte au moins un type de données et certains paramètres acceptent plusieurs types de données. Maintenez le curseur au-dessus du champ de paramètre d'une instruction pour voir les types de données acceptés par un paramètre particulier.

Tableau 4- 3 Types de données pris en charge par le S7-1200

Types de données	Description
Types de données de bit et séquence de bit	<ul style="list-style-type: none">Bool est une valeur booléenne ou de bit.Byte est une valeur d'octet de 8 bits.Word est une valeur de 16 bits.DWord est une valeur mot double de 32 bits.
Types de données en nombres entiers	<ul style="list-style-type: none">USInt (entier 8 bits non signé) et SInt (entier 8 bits signé) sont des entiers "courts" (8 bits ou 1 octet de mémoire) qui peuvent être signés ou non signés.UInt (entier 16 bits non signé) et Int (entier 16 bits signé) sont des entiers (16 bits ou 1 mot de mémoire) qui peuvent être signés ou non signés.UDInt (entier 32 bits non signé) et DInt (entier 32 bits signé) sont des entiers doubles (32 bits ou 1 mot double de mémoire) qui peuvent être signés ou non signés.
Types de données en nombre réel	<ul style="list-style-type: none">Real est un nombre réel 32 bits ou une valeur à virgule flottante.LReal est un nombre réel 64 bits ou une valeur à virgule flottante.

Types de données	Description
Types de données de date et heure	<ul style="list-style-type: none"> Date est une valeur de date de 16 bits (similaire à UInt) qui contient le nombre de jours depuis le 1er janvier 1990. La valeur de date maximale est 65378 (16#FF62), ce qui correspond au 31 juin 2168. Toutes les valeurs Date possibles sont valides. DTL (durée date et heure) est une structure de 12 octets qui mémorise des informations sur la date et l'heure selon une structure prédéfinie. <ul style="list-style-type: none"> Année (UInt) : 1970 à 2554 Mois (USInt) : 1 à 12 Jour (USInt) : 1 à 31 Jour de la semaine (USInt) : 1 (dimanche) à 7 (samedi) Heures (USInt) : 0 à 23 Minutes (USInt) : 0 à 59 Secondes (USInt) : 0 à 59 Nanosecondes (UDInt) : 0 à 999 999 999 Time est une valeur temps CEI 32 bits (similaire à une Dint) qui mémorise le nombre de millisecondes (de 0 à 24 jours 20 heures 31 minutes 23 secondes et 647 ms). Toutes les valeurs Time possibles sont valides. Les valeurs Time peuvent être utilisées pour des calculs et les heures négatives sont possibles. TOD (moment de la journée) est une valeur horaire 32 bits (similaire à une Dint) qui contient le nombre de millisecondes depuis minuit (de 0 à 86 399 999).
Types de données caractère et chaîne	<ul style="list-style-type: none"> Char est un caractère simple de 8 bits. String est une chaîne de longueur variable allant jusqu'à 254 caractères.
Types de données séries et structures	<ul style="list-style-type: none"> Array contient de multiples éléments du même type de données. Il est possible de créer des séries dans les éditeurs d'interface des blocs OB, FC, FB et DB. Vous ne pouvez pas créer de série dans l'éditeur de variables API. Struct définit une structure de données composée d'autres types de données. Le type de données Struct peut être utilisé pour gérer un groupe de données liées à un processus comme une unité de données simple. Vous déclarez le nom et la structure interne des données pour le type de données Struct dans l'éditeur de bloc de données ou un éditeur d'interface de bloc. <p>Les séries et structures peuvent également être assemblées en structures plus grandes. Une structure peut être imbriquée à une profondeur allant jusqu'à huit niveaux. Par exemple, vous pouvez créer une structure de structures contenant des séries.</p>

Types de données	Description
Types de données API	<p>Le type de données API est une structure de données programmable qui définit une structure de données client que vous pouvez utiliser à de multiples reprises dans votre programme. Lorsque vous créez un type de données API, le nouveau type de données API apparaît dans les listes de sélecteur de type de données dans l'éditeur DB et l'éditeur d'interface bloc de code.</p> <p>Les types de données API peuvent être utilisés directement comme type de données dans un interface de bloc de code ou dans des blocs de données.</p> <p>Les types de données API peuvent être utilisés comme modèle pour la création de multiples blocs de données globales utilisant la même structure de données.</p>
Types de données pointeur	<ul style="list-style-type: none"> Pointer fournit une référence indirecte à l'adresse d'une variable. Il occupe 6 octets (48 bits) dans la mémoire et peut inclure les informations suivantes sur une variable : Nombre DB (ou 0 si les données ne sont pas stockées dans une DB), zone de mémoire dans la CPU et l'adresse de la mémoire. Any fournit une référence indirecte au début de la zone de données et identifie sa longueur. Le pointeur Any utilise 10 octets de mémoire et peut inclure les informations suivantes : Le type de données des éléments de données, le nombre d'éléments de données, la zone de mémoire ou le nombre de DB et l'adresse de début "octet.bit" des données. Variant fournit une référence indirecte aux variables des différents types ou paramètres de données. Le pointeur Variant reconnaît les structures et composants individuels de structure. Variant n'occupe aucun espace dans la mémoire.

Bien qu'ils ne soient pas disponibles comme types de données, les formats numériques BCD (décimal codé binaire) sont pris en charge par les opérations de conversion.

- BCD16 est une valeur de 16 bits (-999 à 999).
- BCD32 est une valeur de 32 bits (-9 999 999 à 9 999 999).

4.4.2

Adressage des zones de mémoire

STEP 7 facilite la programmation symbolique. Vous créez des noms symboliques ou "variables" pour les adresses des données, soit sous forme de variables API affectées aux adresses mémoire et E/S, soit sous forme de variables locales utilisées dans un bloc de code. Pour utiliser ces variables dans votre programme utilisateur, il vous suffit d'entrer le nom de la variable comme paramètre de l'instruction. Pour une meilleure compréhension de la manière dont la CPU organise et adresse les zones de mémoire, les paragraphes suivants expliquent l'adressage "absolu" référencé par les variables API. La CPU offre plusieurs possibilités pour stocker les données pendant l'exécution du programme utilisateur :

- Mémoire globale : La CPU fournit diverses zones de mémoire spécialisées, à savoir les entrées (I), les sorties (Q) et les mémentos (M). Cette mémoire est accessible à tous les blocs de code sans restriction.
- Blocs de données (DB) : Vous pouvez inclure des blocs de données (DB) dans votre programme utilisateur afin de sauvegarder les données des blocs de code. Les données sauvegardées sont conservées une fois l'exécution du bloc de code associé achevée. Un DB "global" contient des données pouvant être utilisées par tous les blocs de code alors qu'un DB d'instance contient les données d'un FB spécifique et a une structure correspondant aux paramètres du FB.
- Mémoire temporaire : Lors de l'appel d'un bloc de code, le système d'exploitation de la CPU alloue de la mémoire temporaire - ou locale (L) - utilisable pendant l'exécution de ce bloc. Lorsque l'exécution de ce bloc de code s'achève, la CPU réalloue la mémoire locale pour l'exécution d'autres blocs de code.

Chaque emplacement de mémoire différent a une adresse unique. Votre programme utilisateur utilise ces adresses pour accéder aux informations contenues dans l'emplacement de mémoire.

Les références aux zones de mémoire d'entrée (I) ou de sortie (Q), telles que I0.3 ou Q1.7, permettent d'accéder à la mémoire image. Pour accéder directement à l'entrée ou à la sortie physique, ajoutez ":P" à la référence (par exemple, I0.3:P, Q1.7:P ou "Arret:P").

Le forçage attribue une valeur fixe à une entrée physique (Ix.y:P) ou à une sortie physique (Qx.y:P) uniquement. Pour forcer une entrée ou une sortie, ajouter ":P" à la variable API ou à l'adresse. Pour plus d'informations, référez-vous à "Forçage de variables dans la CPU" (Page 359).

Tableau 4- 4 Zones de mémoire

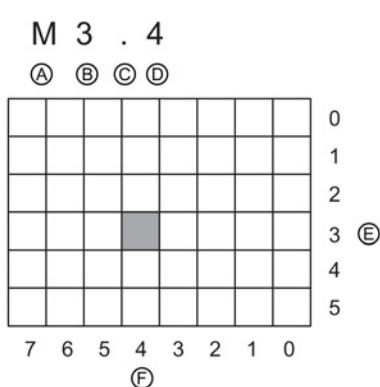
Zone de mémoire	Description	Forçage permanent	Rémanence
I Mémoire image des entrées I_.P ¹ (entrée physique)	Copiée des entrées physiques au début du cycle	Non	Non
	Lecture directe des entrées physiques sur la CPU, le SB et le SM	Oui	Non
Q Mémoire image des sorties Q_.P ¹ (sortie physique)	Copiée dans les sorties physiques au début du cycle	Non	Non
	Écriture directe dans les sorties physiques sur la CPU, le SB et le SM	Oui	Non
M Mémentos	Mémoire de commande et de données	Non	Oui (facultatif)
L Mémoire temporaire	Données temporaires pour un bloc, locales à ce bloc	Non	Non
DB Bloc de données	Mémoire de données ainsi que mémoire de paramètres pour les FB	Non	Oui (facultatif)

¹ Pour accéder directement aux entrées et sorties physiques (ou pour les forcer), ajoutez ":P" à l'adresse ou à la variable (par exemple, I0.3:P, Q1.7:P ou "Arret:P").

Chaque emplacement de mémoire différent a une adresse unique. Votre programme utilisateur utilise ces adresses pour accéder aux informations contenues dans l'emplacement de mémoire. L'adresse absolue est constituée des éléments suivants :

- Zone de mémoire (comme I, Q ou M)
- Taille des données accessibles (comme "B" pour Byte ou "W" pour Word)
- Adresse des données (comme Byte 3 ou Word 3)

Lors de l'accès à un bit dans l'adresse pour une valeur booléenne, vous ne devez pas entrer de code mnémonique pour la dimension. Vous saisissez uniquement la zone de mémoire, l'emplacement d'octet et l'emplacement de bit pour les données (comme I0.0, Q0.1 ou M3.4).

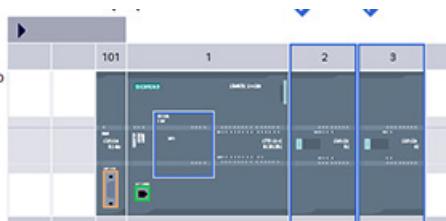


Adresse absolue d'une zone de mémoire :

- A Identificateur de zone de mémoire
- B Adresse d'octet : Octet 3
- C Séparateur ("octet.bit")
- D Position du bit dans l'octet (bit 4 sur 8)
- E Octets de la zone de mémoire
- F Bits de l'octet sélectionné

Dans l'exemple, la zone de mémoire et l'adresse d'octet (M = zone de mémoire de bit et 3 = Byte 3) sont suivies par un point (".") pour séparer l'adresse du bit (bit 4).

Configuration des E/S dans la CPU et les modules E/S



Vue d'ensemble des appareils					
Module	Empla...	Adresse E	Adresse S	Type	N° de r...
103					
102					
RS485_1	101			CM 1241 (RS485)	6ES7
PLC_1	1			CPU 1214C D/O/O	6ES7
DI16/DO10	1.1	0...1	0...1	DI16/DO10	
	1.2	64...67		AI2	
AO1 x 12Bit	1.3		80...81	Signal Board AO1	6ES7
HSC_1	1.16	1000.....		Compteur rapide (...	
HSC_2	1.17			Compteur rapide (...	
HSC_3	1.18			Compteur rapide (...	
HSC_4	1.19			Compteur rapide (...	
HSC_5	1.20			Compteur rapide (...	
HSC_6	1.21			Compteur rapide (...	
Pulse_1	1.32			Générateur d'impu...	
Pulse_2	1.33			Générateur d'impu...	
Interface P...	X1			Interface PROFINET	
DI8 x DC24V...	2		8	SM 1221 DI8 x DC...	6ES7

Lorsque vous ajoutez une CPU et des modules d'E/S à votre configuration d'appareil, STEP 7 affecte automatiquement les adresses I et Q. Vous pouvez modifier l'adressage par défaut en sélectionnant le champ d'adresse dans la configuration d'appareil et en entrant de nouveaux nombres.

- STEP 7 attribue les entrées et sorties numériques dans des groupes de 8 points (1 octet), que le module utilise tous les points ou non.
- STEP 7 affecte les entrées et sorties numériques dans des groupes de 2, où chaque point analogique occupe 2 octets (16 bits).

La figure montre un exemple de CPU 1214C avec deux SM et un SB. Dans cet exemple, vous pourriez modifier l'adresse du module DI8 à 2 au lieu de 8. L'outil vous aide en modifiant les plages d'adresse qui ne sont pas de la bonne dimension ou qui sont en conflit avec d'autres adresses.

4.4.3

Accès à une "tranche" d'un type de données de variable

Vous pouvez accéder aux variables API et aux variables de blocs de données au niveau bit, octet ou mot selon leur taille. Voici la syntaxe pour accéder à une telle tranche de données :

- "<nom variable API>".xn (accès bit)
- "<nom variable API>".bn (accès octet)
- "<nom variable API>".wn (accès mot)
- "<nom bloc de données>.<nom variable>.xn (accès bit)
- "<nom bloc de données>.<nom variable>.bn (accès octet)
- "<nom bloc de données>.<nom variable>.wn (accès mot)

Il est possible d'accéder à une variable double mot par les bits 0 à 31, les octets 0 à 3 ou les mots 0 et 1. Il est possible d'accéder à une variable mot par les bits 0 à 15, les octets 0 à 1 ou le mot 0. Il est possible d'accéder à une variable octet par les bits 0 à 7 ou l'octet 0. Les tranches bit, octet et mot peuvent être utilisées partout où on attend des bits, octets ou mots comme opérandes.

BYTE																															
WORD																WORD															
DWORD																															
x31	x30	x29	x28	x27	x26	x25	x24	x23	x22	x21	x20	x19	x18	x17	x16	x15	x14	x13	x12	x11	x10	x9	x8	x7	x6	x5	x4	x3	x2	x1	x0
b3								b2								b1								b0							
w1																w0															

Remarque

Les types de données auxquels il est possible d'accéder par tranche sont les types Byte, Char, Conn_Any, Date, DInt, DWord, Event_Any, Event_Att, Hw_Any, Hw_Device, HW_Interface, Hw_Io, Hw_Pwm, Hw_SubModule, Int, OB_Any, OB_Att, OB_Cyclic, OB_Delay, OB_WHINT, OB_PCYCLE, OB_STARTUP, OB_TIMEERROR, OB_Tod, Port, Rtm, SInt, Time, Time_Of_Day, UDInt, UInt, USInt et Word. Il est possible d'accéder aux variables API de type Real par tranche, mais pas aux variables de blocs de données de type Real.

Exemples

Dans la table de variables API, "DW" est une variable déclarée de type DWORD. Les exemples suivants montrent l'accès à cette variable par tranche de bit, octet et mot :

	CONT	LOG	SCL
Accès bit	"DW".x11 	"DW".x11 	IF "DW".x11 THEN ... END_IF;
Accès octet	"DW".b2 == Byte "DW".b3 	"DW".b2 == "DW".b3 Byte "DW".b2 — IN1 "DW".b3 — IN2 	IF "DW".b2 = "DW".b3 THEN ... END_IF;
Accès mot			out := "DW".w0 AND "DW".w1;

4.4.4 Accès à une variable par un type de données ajouté AT

L'ajout d'un type de données à une variable via AT vous permet d'accéder à une variable déjà déclarée d'un bloc à accès standard avec une déclaration de type de données différent ajoutée. Vous pouvez, par exemple, accéder aux bits individuels d'une variable de type Byte, Word ou DWord au moyen d'un tableau de booléens (Array of Bool).

Déclaration

Pour ajouter un type de données à un paramètre, déclarez un paramètre supplémentaire directement après le paramètre concerné et sélectionnez le type de données "AT". L'éditeur crée l'ajout et vous pouvez alors choisir le type de données, la structure ou le tableau que vous voulez utiliser comme type de données ajouté.

Exemple

Cet exemple montre les paramètres d'entrée d'un FB à accès standard. On ajoute comme type de données à la variable B1 de type octet un tableau de booléens :

01	■	B1	Byte	0.0
01	■	OV	AT "B1"	Array[0..7] of Bool
01	■	OV[0]	Bool	0.0
01	■	OV[1]	Bool	0.1
01	■	OV[2]	Bool	0.2
01	■	OV[3]	Bool	0.3
01	■	OV[4]	Bool	0.4
01	■	OV[5]	Bool	0.5
01	■	OV[6]	Bool	0.6
01	■	OV[7]	Bool	0.7

Dans cet autre exemple, une variable de type DWord reçoit comme nouveau type de données une structure, qui comprend un mot, un octet et deux opérateurs logiques (booléens) :

01	■	DW1	DWord	2.0
01	■	DW1_Struct	AT "DW1"	Struct
01	■	W1	Word	0.0
01	■	B1	Byte	2.0
01	■	BO1	Bool	3.0
01	■	BO2	Bool	3.1

La colonne Décalage de l'interface de bloc montre l'emplacement des types de données ajoutés par rapport à la variable initiale.

Vous pouvez adresser les types de données ajoutés directement dans la logique du programme :

CONT	LOG	SCL
#OV[1]		IF #OV[1] THEN ... END_IF;
#DW1_Struct.W1 == Word W#16#000C		IF #DW1_Struct.W1 = W#16#000C THEN ... END_IF;
MOVE #DW1_Struct.B1 — IN * OUT1 —		out1 := #DW1_Struct.B1;
#OV[4] #DW1_Struct.BO2		IF #OV[4] AND #DW1_Struct.BO2 THEN ... END_IF;

Règles

- L'ajout d'un type de données à des variables n'est possible que dans les FB et FC à accès standard (non optimisé).
- Vous pouvez ajouter un type de données à des paramètres pour tous les types de blocs et toutes les sections de déclaration.
- Vous pouvez utiliser un paramètre auquel on a ajouté un type de données comme n'importe quel autre paramètre du bloc.
- Vous ne pouvez pas ajouter de type de données aux paramètres de type VARIANT.
- La taille du paramètre d'ajout doit être inférieure ou égale à la taille du paramètre auquel on ajoute le type de données.
- Vous devez déclarer la variable d'ajout de type de données immédiatement après la variable à laquelle on ajoute le type de données et sélectionner le mot-clé "AT" comme sélection initiale du type de données.

4.5

Sorties d'impulsions

La CPU ou le Signal Board (SB) peut être configuré(e) pour fournir quatre générateurs d'impulsion afin de contrôler les fonctions de sortie d'impulsions rapides, soit en tant que modulation de largeur d'impulsion (PWM), soit en tant que sortie de train d'impulsion (PTO). Les opérations de déplacement basiques utilisent les sorties PTO. Vous pouvez attribuer chaque générateur d'impulsion soit au PWM, soit au PTO, mais pas aux deux à la fois.



Les sorties d'impulsion ne peuvent pas être utilisées par d'autres opérations dans le programme utilisateur. Lorsque vous configurez les sorties de la CPU ou du SB comme générateurs d'impulsions, les adresses de sortie correspondantes sont supprimées de la mémoire Q et ne peuvent pas être utilisées à d'autres fins dans votre programme utilisateur. Si votre programme utilisateur écrit une valeur dans une sortie utilisée comme générateur d'impulsions, la CPU n'écrit pas cette valeur dans la sortie physique.

Remarque

Ne dépasser pas la fréquence d'impulsion maximale.

La fréquence d'impulsion maximale des générateurs de sortie d'impulsions est de 1 MHz pour la CPU 1217C et de 100 kHz pour les CPU 1211C, 1212C, 1214C et 1215C ; de 20 kHz (pour un SB standard) ; ou de 200 kHz (pour un SB rapide).

Les quatre générateurs d'impulsions ont des affectations E/S par défaut ; toutefois, ils peuvent être configurés sur n'importe quelle sortie TOR sur la CPU ou le SB. On ne peut pas affecter de générateurs d'impulsions sur la CPU à des E/S distribuées.

Lors de la configuration des opérations de déplacement basiques, rappelez vous que STEP 7 ne vous alerte **pas** si vous configurez un axe avec une vitesse ou une fréquence maximale qui dépasse cette limite matérielle. Afin d'éviter tout problème avec votre application, assurez-vous toujours de ne pas dépasser la fréquence d'impulsions maximale du matériel.

Vous pouvez utiliser les sorties CPU intégrées ou les sorties du Signal Board optionnel. Les numéros des sorties sont indiqués dans le tableau suivant (sur la base de la configuration par défaut des sorties). Si vous avez modifié la numérotation des sorties, les numéros des sorties seront ceux que vous avez définis. Notez que PWM ne nécessite qu'une sortie alors que PTO peut, en option, utiliser deux sorties par voie. Si une sortie n'est pas utilisée pour une fonction d'impulsion, elle est disponible pour d'autres usages.

Les quatre générateurs d'impulsions ont des affectations E/S par défaut ; toutefois, ils peuvent être configurés sur n'importe quelle sortie TOR sur la CPU ou le SB. On ne peut pas affecter de générateurs d'impulsions à des SM ou à des E/S distribuées.

Tableau 4- 5 Affectations par défaut des sorties aux générateurs d'impulsions

Description	Impulsion	Sens
PTO1		
E/S intégrées	Q0.0	Q0.1
E/S du SB	Q4.0	Q4.1
PWM1		
Sorties intégrées	Q0.0	-
Sorties du SB	Q4.0	-
PTO2		
E/S intégrées	Q0.2	Q0.3
E/S du SB	Q4.2	Q4.3
PWM2		
Sorties intégrées	Q0.2	-
Sorties du SB	Q4.2	-
PTO3		
E/S intégrées	Q0.4 ¹	Q0.5 ¹
E/S du SB	Q4.0	Q4.1
PWM3		
Sorties intégrées	Q0.4 ¹	-
Sorties du SB	Q4.1	-
PTO4		
E/S intégrées	Q0.6 ²	Q0.7 ²
E/S du SB	Q4.2	Q4.3
PWM4		
Sorties intégrées	Q0.6 ²	-
Sorties du SB	Q4.3	-

¹ La CPU 1211C ne comporte pas de sorties Q0.4, Q0.5, Q0.6 et Q0.7. Ces sorties ne peuvent donc pas être utilisées dans la CPU 1211C.

² La CPU 1212C ne comporte pas de sorties Q0.6 et Q0.7. Ces sorties ne peuvent donc pas être utilisées dans la CPU 1212C.

³ Ce tableau s'applique aux fonctions PTO/PWM des CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C et CPU 1217C.

Création facile de la configuration des appareils

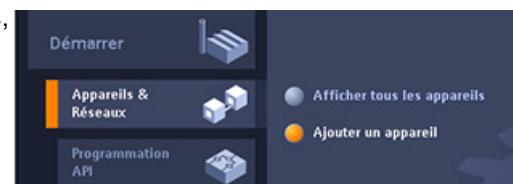
Vous créez la configuration des appareils pour votre automate en ajoutant une CPU et des modules supplémentaires à votre projet.



- ① Module de communication (CM) ou processeur de communication (CP) : jusqu'à 3, insérés aux emplacements 101, 102 et 103
- ② CPU : emplacement 1
- ③ port Ethernet de la CPU
- ④ Signal Board (SB), Communication Board (CB) ou Battery Board (BB) : au plus 1, inséré dans la CPU
- ⑤ Module d'entrées-sorties (SM) TOR ou analogiques : jusqu'à 8, insérés aux emplacements 2 à 9
(les CPU 1214C, CPU 1215C et CPU 1217C en autorisent 8, la CPU 1212C en autorise 2, la CPU 1211C n'en autorise aucun)

Pour créer la configuration des appareils, ajoutez un appareil à votre projet.

- Dans la vue du portail, sélectionnez "Appareils & réseaux" et cliquez sur "Ajouter un appareil".
- Dans la vue du projet, double-cliquez sur "Ajouter un appareil" sous le nom de projet.



5.1 Chargement de la configuration d'une CPU connectée

STEP 7 offre deux méthodes de chargement de la configuration matérielle d'une CPU connectée :

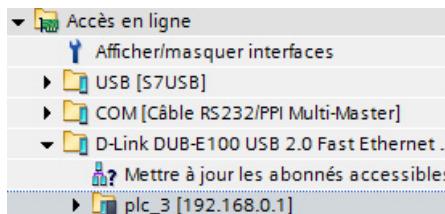
- Chargement de l'appareil connecté comme un nouveau poste
- Configuration d'une CPU non spécifiée et détection de la configuration matérielle de la CPU connectée

À noter, toutefois, que la première méthode charge à la fois la configuration matérielle et le logiciel de la CPU connectée.

Chargement d'un appareil comme un nouveau poste

Pour charger un appareil connecté comme un nouveau poste, suivez les étapes ci-dessous :

1. Affichez votre interface de communication depuis le nœud "Accès en ligne" de l'arborescence du projet.
2. Double-cliquez sur "Mettre à jour les abonnés accessibles".
3. Sélectionnez l'API dans les appareils détectés.



4. Dans le menu En ligne de STEP 7, sélectionnez la commande de menu "Charger l'appareil comme un nouveau poste (matériel et logiciel)".

STEP 7 charge à la fois la configuration matérielle et les blocs de programme.

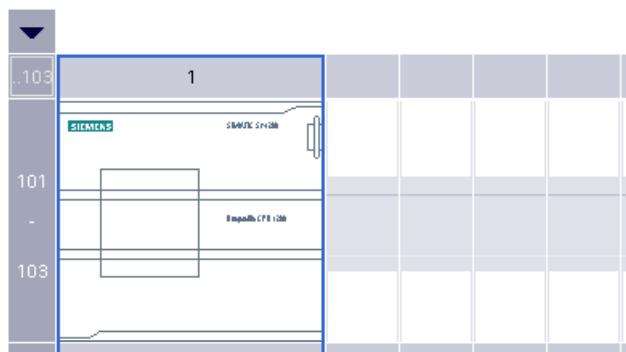
Détection de la configuration matérielle d'une CPU non spécifiée



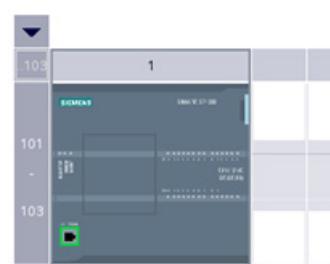
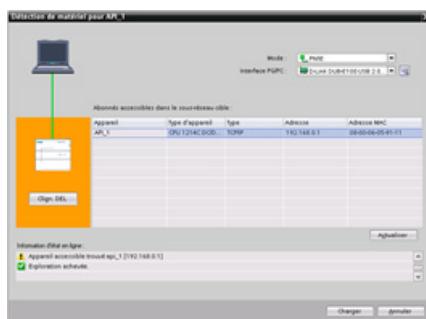
Lorsque vous êtes connecté à une CPU, vous pouvez charger la configuration de cette CPU, y compris tous les modules, dans votre projet. Il vous suffit de créer un nouveau projet et de sélectionner la "CPU non spécifiée" au lieu d'une CPU spécifique (vous pouvez également ne pas passer par la configuration matérielle en sélectionnant "Créer un programme API" dans "Mise en route". STEP 7 crée ensuite automatiquement une CPU non spécifiée.)

Dans l'éditeur de programmes, vous sélectionnez la commande "Détection du matériel" du menu "En ligne".

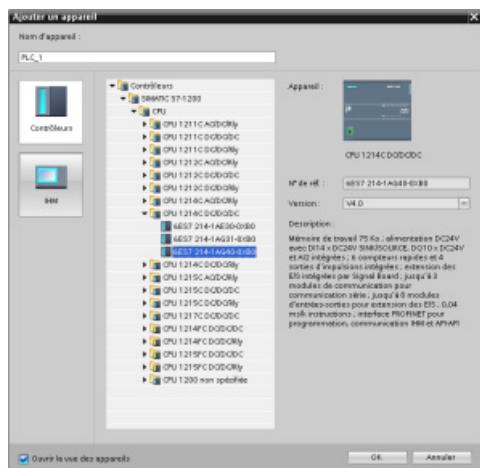
Dans l'éditeur de configuration des appareils, vous sélectionnez l'option de détection de configuration de l'appareil connecté.



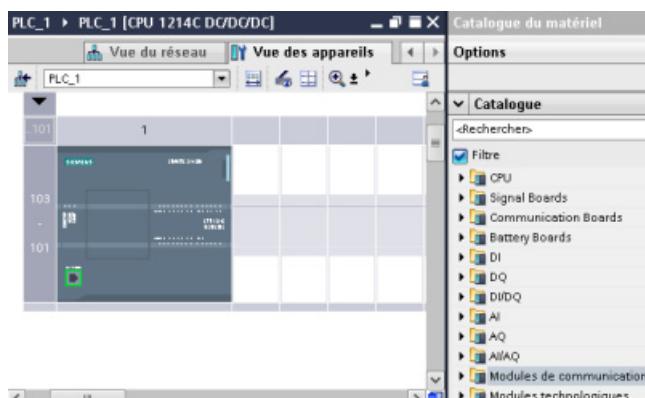
Une fois que vous avez sélectionné la CPU dans la boîte de dialogue en ligne et cliqué sur le bouton Charger, STEP 7 télécharge la configuration matérielle depuis la CPU, en incluant tous les modules (SM, SB ou CM). Vous pouvez alors configurer les paramètres de la CPU et des modules (Page 82).



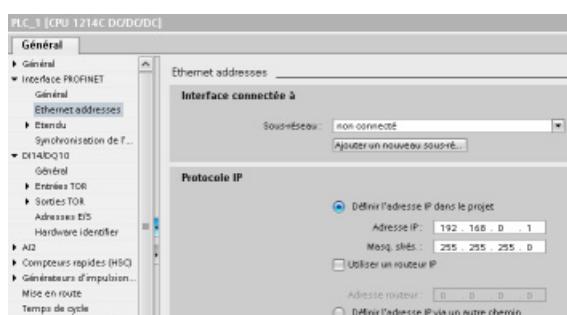
5.2 Ajout d'une CPU à la configuration



Vous créez votre configuration d'appareil en insérant une CPU dans votre projet. Sélectionnez la CPU dans la boîte de dialogue "Ajouter un appareil" et cliquez sur "OK" pour ajouter la CPU au projet.



La vue des appareils affiche la CPU et le châssis.



Selectionner la CPU dans la vue des appareils provoque l'affichage des propriétés de la CPU dans la fenêtre d'inspection. Utilisez ces propriétés pour configurer les paramètres de fonctionnement de la CPU (Page 82).

Remarque

La CPU ne dispose pas d'une adresse IP préconfigurée. Vous devez affecter manuellement une adresse IP à la CPU pendant la configuration des appareils. Si votre CPU est connectée à un routeur sur le réseau, vous entrez aussi l'adresse IP du routeur.

5.3

Changer d'appareil

Vous pouvez changer le type d'appareil d'une CPU ou d'un module configuré. Depuis Configuration de l'appareil, cliquez sur l'appareil avec le bouton droit de la souris et choisissez la commande "Remplacer l'appareil" dans le menu contextuel. Depuis la boîte de dialogue, naviguez jusqu'à la CPU ou au module que vous souhaitez remplacer puis sélectionnez-le. La boîte de dialogue Changer l'appareil vous fournit des informations sur la compatibilité entre les deux appareils.

Remarque

Remplacement de l'appareil : remplacer votre CPU V3.0 par une CPU V4.1

Vous pouvez ouvrir un projet STEP 7 V12 dans STEP 7 V13 et remplacer les CPU V3.0 par des CPU V4.1. Vous ne pouvez pas remplacer les CPU issues de versions antérieures à la V3.0. Quand vous remplacez une CPU V3.0 par une CPU V4.1, tenez compte des différences (Page 453) dans les fonctions et du comportement des deux versions, et des actions que vous devez entreprendre.

Si vous avez un projet pour une version de CPU antérieure à la V3.0, vous devez d'abord mettre la CPU à niveau vers la V3.0 et ensuite la mettre à niveau vers la V4.1.

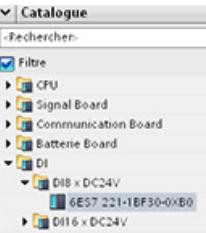
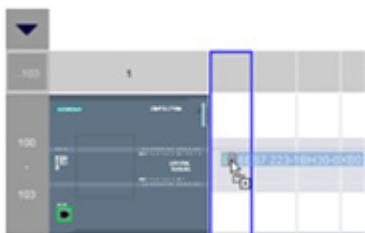
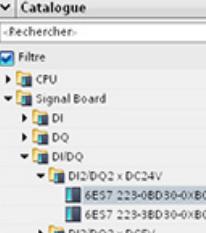
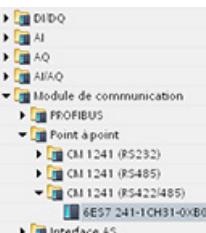
5.4 Ajout de modules à la configuration

Utiliser le catalogue de matériel pour ajouter des modules à la CPU :

- Le module d'entrées-sorties (SM) fournit des points d'E/S numériques ou analogiques supplémentaires. Ces modules se raccordent sur le côté droit de la CPU.
- Le Signal Board (SB) fournit seulement quelques points d'E/S supplémentaires pour la CPU. Le SB s'installe à l'avant de la CPU.
- Le Battery Board 1297 fournit une sauvegarde à long terme de l'horloge temps réel. Il s'installe à l'avant de la CPU.
- Le Communication Board (CB) fournit un port de communication supplémentaire (comme RS485). Le Communication Board s'installe à l'avant de la CPU.
- Le module de communication (CM) et le processeur de communication (CP) fournissent un port de communication supplémentaire, comme PROFIBUS ou GPRS. Ces modules se raccordent sur le côté gauche de la CPU.

Pour insérer un module dans la configuration des appareils, sélectionnez le module concerné dans le catalogue du matériel et double-cliquez ou faites glisser le module dans l'emplacement mis en évidence. Pour que les modules soient opérationnels, vous devez les ajouter à la configuration des appareils et charger dans la CPU la configuration matérielle pour les modules.

Tableau 5- 1 Ajouter un module à la configuration de l'appareil

Module	Sélectionnez le module	Insérez le module	Résultat
SM			
SB, BB ou CB			
CM ou CP			

Grâce à la fonction "commande de configuration" (Page 81), vous pouvez ajouter des modules d'entrées-sorties et des signal boards à votre configuration d'appareil qui pourraient ne pas correspondre au matériel réel d'une application spécifique, mais qui seront utilisés dans des applications associées qui partagent un programme utilisateur commun, un modèle de CPU et peut-être certains des modules configurés.

5.5 Commande de configuration

La commande de configuration peut être un outil utile si vous créez une solution (machine) automatisée que vous avez l'intention d'utiliser dans plusieurs installations avec des variations.

La commande de configuration avec STEP 7 et le S7-1200 vous permettent de définir une configuration maximale pour une machine standard et d'utiliser des versions (options) utilisant un sous-ensemble de cette configuration. Le manuel PROFINET avec STEP 7 (<http://support.automation.siemens.com/WW/view/en/49948856>) nomme ces types de projets "projets de machine standard".

Vous pouvez charger une configuration d'appareil STEP 7 et un programme d'utilisateur dans différentes configurations d'API installées. Vous devez seulement procéder à quelques adaptations simples pour faire correspondre le projet STEP 7 à l'installation réelle.

Un enregistrement des données de commande que vous programmez dans le bloc de programme de démarrage signale à la CPU les modules qui manquent dans l'installation réelle, en comparaison avec la configuration, et les modules qui sont situés à des emplacements différents, en comparaison avec la configuration. La commande de configuration n'a pas d'impact sur le paramétrage des modules.

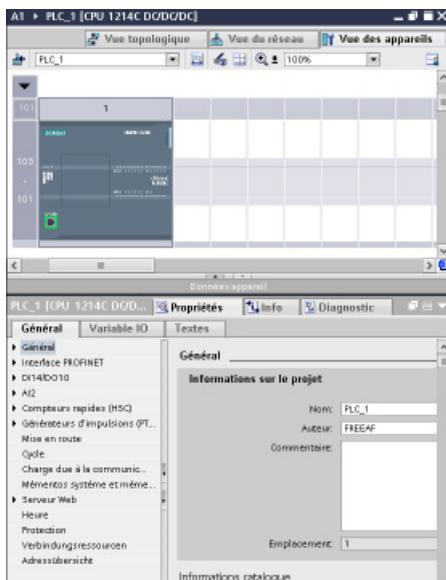
La commande de configuration vous donne la flexibilité de modifier l'installation, dans la mesure où la configuration réelle dérive de la configuration maximale de l'appareil dans STEP 7.

Vous pouvez trouver des instructions et des exemples de commande de configuration dans le *Manuel système de l'automate programmable S7-1200*.

5.6

Configuration du fonctionnement de la CPU et des modules

Pour configurer les paramètres de fonctionnement de la CPU, sélectionnez la CPU dans la vue des appareils et servez-vous de l'onglet "Propriétés" de la fenêtre d'inspection.



Vous pouvez configurer les propriétés de la CPU suivantes :

- Adresse PROFINET IP et synchronisation d'horloge pour la CPU
- Comportement au démarrage de la CPU suite à une transition d'alimentation OFF vers ON
- E/S TOR et analogiques locales (intégrées), compteurs rapides (HSC) et générateurs d'impulsions
- Horloge système (heure, zone horaire et heure d'été)
- Protection en lecture/écriture et mot de passe d'accès à la CPU
- Temps de cycle maximum ou temps de cycle minimum fixé et charge due à la communication
- Propriétés du serveur Web

Configuration du fonctionnement ARRET-MARCHE de la CPU

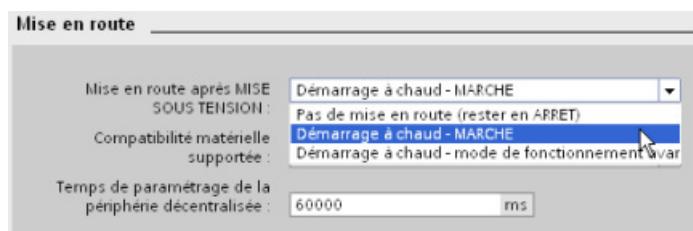
A chaque fois que l'état de fonctionnement passe de ARRET à MARCHE, la CPU efface la mémoire image des entrées, initialise la mémoire image des sorties et traite les OB de démarrage. Ainsi, tout accès en lecture à la mémoire image des entrées par des instructions dans les OB de démarrage lira zéro et non l'état en cours de l'entrée physique. Pour lire l'état en cours d'une entrée physique pendant la mise en route, vous devez effectuer une lecture directe. Les OB de démarrage ainsi que tous les FB et FC associés sont ensuite exécutés. En présence de plus d'un OB de démarrage, chacun est exécuté par ordre de numéro d'OB, l'OB de plus petit numéro étant exécuté en premier.

La CPU exécute également les tâches suivantes pendant le traitement de mise en route.

- Les alarmes sont mises en file d'attente mais ne sont pas traitées pendant la phase de mise en route.
- Aucune surveillance du temps de cycle n'est effectuée pendant la phase de mise en route.
- Il est possible de modifier la configuration de HSC (compteur rapide), PWM (modulation de largeur des impulsions) et des modules PtP (communication point à point) pendant la mise en route.
- L'exécution réelle de HSC, PWM et des modules de communication point à point n'est possible qu'à l'état MARCHE.

Une fois l'exécution des OB de démarrage achevée, la CPU passe à l'état MARCHE et traite les tâches de commande lors d'un cycle continu.

Utiliser les propriétés de la CPU pour configurer la façon dont la CPU démarre après un cycle d'alimentation.



- En mode ARRÊT
- En mode MARCHE
- Dans le mode précédent (avant le cycle d'alimentation)

La CPU effectue un démarrage à chaud avant de passer à l'état MARCHE. Un démarrage à chaud réinitialise toute la mémoire non rémanente à ses valeurs initiales par défaut, mais la CPU conserve les valeurs en cours stockées en mémoire rémanente.

Remarque

La CPU effectue toujours un redémarrage après un chargement.

Lorsque vous chargez un élément de votre projet (tel qu'un bloc de programme, un bloc de données ou la configuration matérielle) dans la CPU, la CPU effectue un redémarrage lors du passage suivant à l'état MARCHE. En plus de l'effacement des entrées, de l'initialisation des sorties et de l'initialisation de la mémoire non rémanente, le redémarrage entraîne également l'initialisation des zones de mémoire rémanentes.

Après le redémarrage consécutif à un chargement, toutes les transitions suivantes de ARRÊT à MARCHE provoquent un démarrage à chaud (qui n'initialise pas la mémoire rémanente).

5.6.1 Le mémento système et le mémento de cadence mettent à disposition des fonctionnalités standard

Vous vous servez des propriétés CPU pour activer les octets "mémento système" et "mémento de cadence". La logique de votre programme peut faire référence aux bits individuels de ces fonctions par leur nom de variable.

- Vous pouvez affecter un octet dans la zone de mémoire M au mémento système. L'octet de mémento système fournit les quatre bits suivants auxquels votre programme utilisateur peut faire référence par les noms de variables suivants :
 - Premier cycle (nom de variable "FirstScan") : Ce bit est mis à 1 pendant la durée du premier cycle une fois l'OB de démarrage terminé. Une fois l'exécution du premier cycle achevée, le bit "Premier cycle" est mis à 0.
 - Etat de diagnostic modifié : (Nom de la variable : "DiagStatusUpdate") : Ce bit est mis à 1 pendant un cycle après que la CPU a consigné un événement de diagnostic. Comme la CPU ne définit pas le bit "DiagStatusUpdate" avant la fin de la première exécution des OB de cycle de programme, votre programme utilisateur ne peut pas détecter s'il y a eu une modification du diagnostic pendant l'exécution des OB de démarrage ou la première exécution des OB de cycle de programme.
 - Toujours 1 (high) (nom de variable "AlwaysTRUE") : Ce bit est toujours à 1.
 - Toujours 0 (low) (nom de variable "AlwaysFALSE") : Ce bit est toujours à 0.
- Vous pouvez affecter un octet dans la zone de mémoire M au mémento de cadence. Chaque bit de l'octet configuré comme mémento de cadence génère une impulsion en signaux carrés. L'octet de mémento de cadence fournit 8 fréquences différentes, de 0,5 Hz (lent) à 10 Hz (rapide). Vous pouvez utiliser ces bits comme bits de commande, en particulier en combinaison avec des instructions sur front, pour déclencher des actions dans le programme utilisateur sur une base cyclique.

La CPU initialise ces octets lors du passage de l'état ARRET à l'état MISE EN ROUTE. Les bits du mémento de cadence changent de manière synchrone avec l'horloge CPU aux états MISE EN ROUTE et MARCHE.



PRUDENCE

Risques liés à l'écrasement des bits du mémento système ou du mémento de cadence

L'écrasement des bits du mémento système ou du mémento de cadence peut altérer les données dans ces fonctions et provoquer un fonctionnement incorrect du programme utilisateur, pouvant entraîner la mort et des blessures graves.

Comme le mémento de cadence et le mémento système ne sont pas réservés en mémoire M, des instructions ou des tâches de communication peuvent écrire dans ces adresses et altérer les données.

Evitez d'écrire des données dans ces adresses afin de garantir le bon fonctionnement de ces fonctions et prévoyez toujours un circuit d'arrêt d'urgence pour votre processus ou votre installation.

La fonction mémento système configure un octet dont les bits sont activés (valeur = 1) pour un événement spécifique.

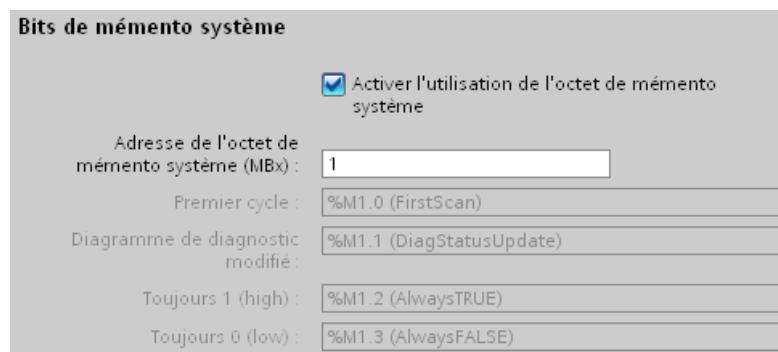


Tableau 5- 2 Mémento système

7	6	5	4	3	2	1	0
Réservés		Toujours désactivé		Toujours activé		Indicateur Etat de diagnostic	Indicateur Premier cycle
Valeur 0		Valeur 0		Valeur 1		<ul style="list-style-type: none"> • 1: Modification • 0: Pas de modification 	<ul style="list-style-type: none"> • 1: Premier cycle après la mise en route • 0: Pas le premier cycle

La fonction mémento de cadence configure un octet qui met les différents bits à 1 et à 0 à intervalles fixes. Chaque bit de cadence génère une impulsion en signaux carrés sur le bit M correspondant. Ces bits peuvent être utilisés comme bits de commande, en particulier en combinaison avec des instructions sur front, pour déclencher des actions dans le code utilisateur sur une base cyclique.

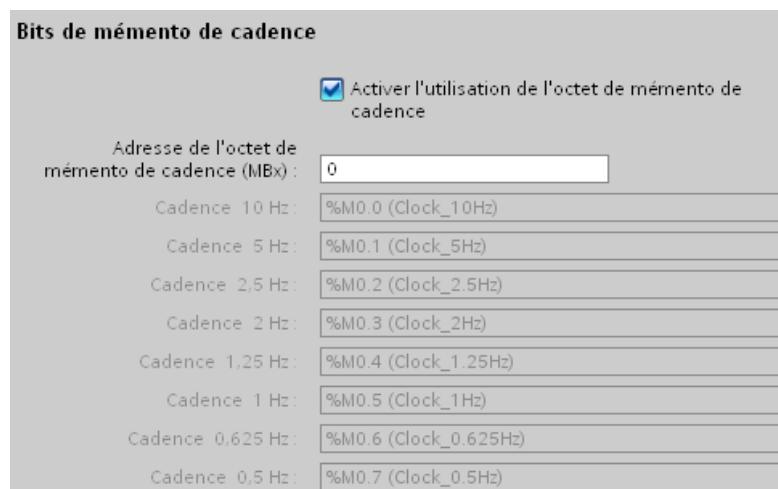


Tableau 5- 3 Mémento de cadence

Numéro du bit	7	6	5	4	3	2	1	0
Nom de variable								
Période (s)	2.0	1.6	1.0	0.8	0.5	0.4	0.2	0.1
Fréquence (Hz)	0.5	0.625	1	1.25	2	2.5	5	10

Comme le mémento de cadence opère de manière asynchrone par rapport au cycle de la CPU, son état peut changer plusieurs fois au cours d'un cycle long.

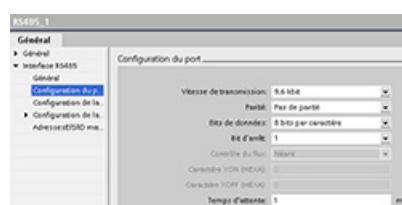
Configuration du fonctionnement des E/S et des modules de communication

Pour configurer les paramètres de fonctionnement d'un module d'entrées/sorties (SM), d'un Signal Board (SB) ou d'un module de communication, sélectionnez le module dans la Vue des appareils et servez-vous de l'onglet "Propriétés" de la fenêtre d'inspection.



Module d'entrées-sorties (SM) et Signal Board (SB)

- **E/S TOR** : Permet de configurer les entrées individuelles, comme pour la détection de Front et "capture d'impulsion" (pour rester allumé ou éteint pour un cycle après une impulsion momentanée élevée ou faible). Permet de configurer les sorties afin qu'elles se figent ou utilisent une valeur de remplacement en cas de passage de l'état MARCHE à l'état ARRET.
- **E/S analogiques** : Permet de configurer les paramètres pour les entrées individuelles (comme la tension/le courant, la portée et le filtrage) et permet également les diagnostics de soupassement ou de dépassement. Configurer les paramètres pour les sorties analogiques individuelles et activer les diagnostics, tels que le court-circuit (pour les sorties de tension) ou les valeurs de débordement.
- **Adresses E/S** : Permet de configurer l'adresse de début pour le jeu d'entrées et de sorties du module.



Module de communication (CM) et carte de communication (CB)

- **Configuration du port** : Permet de configurer les paramètres de communication, tels que la vitesse de transmission, la parité, les bits de données, les bits d'arrêt et le temps d'attente.
- **Emission et réception de messages** : Permet de configurer les options liées à l'émission et la réception de données (les paramètres de début et de fin de message, par exemple).

Vous pouvez également modifier ces paramètres de configuration avec votre programme utilisateur.

5.7

Configuration de l'adresse IP de la CPU

Etant donné que la CPU ne possède pas d'adresse IP préconfigurée, vous devez affecter manuellement une adresse IP. Vous configurez l'adresse IP et les autres paramètres de l'interface PROFINET en même temps que les propriétés de la CPU.

- Dans un réseau PROFINET, une adresse MAC (Media Access Control) unique est affectée par le fabricant à chaque appareil, afin d'en permettre l'identification. Chaque appareil doit également avoir une adresse IP.
- Un sous-réseau est un groupement logique d'appareils réseau connectés. Un masque (également appelé masque de sous-réseau ou masque de réseau) définit les frontières d'un sous-réseau. La seule liaison possible entre différents sous-réseaux se fait via un routeur. Les routeurs constituent le lien entre les LAN et utilisent les adresses IP pour transmettre et recevoir des paquets de données.

Avant de pouvoir télécharger une adresse IP vers la CPU, vous devez vous assurer que l'adresse IP pour votre CPU est compatible avec l'adresse IP de votre console de programmation.

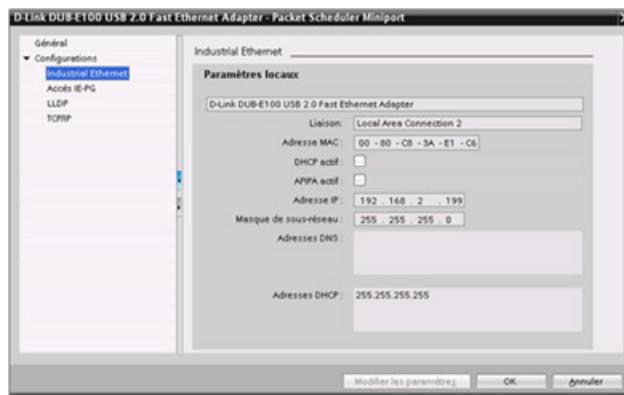
Vous pouvez utiliser STEP 7 pour déterminer l'adresse IP de votre console de programmation :

1. Développez le dossier "Accès en ligne" dans l'arborescence du projet pour afficher vos réseaux.
2. Sélectionnez le réseau qui se connecte à la CPU.
3. Faites un clic droit sur le réseau spécifique pour afficher le menu contextuel.
4. Sélectionnez la commande "Propriétés".

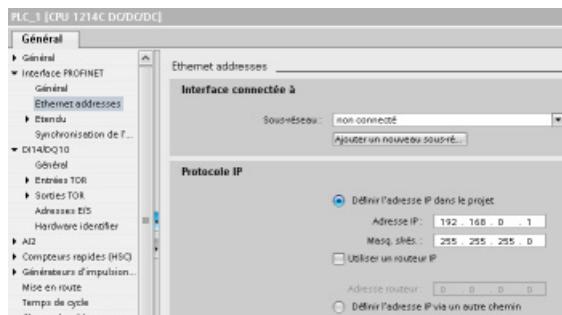
Remarque

L'adresse IP de la CPU doit être compatible avec l'adresse IP et le masque de sous-réseau de la console de programmation. Consultez votre spécialiste réseau pour une adresse IP et un masque de sous-réseau adaptés à votre CPU.

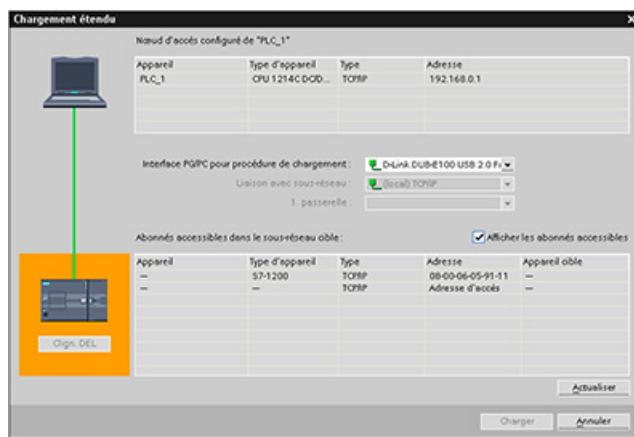
5.7 Configuration de l'adresse IP de la CPU



La fenêtre "Propriétés" affiche les réglages pour la console de programmation.



Après avoir déterminé l'adresse IP et le masque de sous-réseau de la CPU, entrez l'adresse IP de la CPU et du routeur (en présence de celui-ci). Pour plus d'informations, référez-vous au *Manuel système de l'automate programmable S7-1200*.



Une fois la configuration achevée, chargez le projet dans la CPU.

Les adresses IP de la CPU et du routeur (en présence de celui-ci) sont configurées lors du chargement du projet.

5.8 La protection d'accès à la CPU ou au bloc de code est facile

La CPU fournit quatre niveaux de sécurité permettant de limiter l'accès à des fonctions spécifiques. Lorsque vous configurez le niveau de sécurité et le mot de passe pour une CPU, vous limitez les fonctions et les zones de mémoire qui sont accessibles sans saisie d'un mot de passe.

Chaque niveau permet d'accéder à certaines fonctions sans mot de passe. Par défaut, il n'y a pas de restriction d'accès ni de protection par mot de passe pour la CPU. Pour limiter l'accès à une CPU, vous configurez les propriétés de la CPU et vous entrez le mot de passe.

L'entrée du mot de passe via un réseau ne met pas en cause la protection par mot de passe de la CPU. La protection par mot de passe ne concerne pas l'exécution des instructions du programme utilisateur, fonctions de communication comprises. La saisie du mot de passe correct permet d'accéder à toutes les fonctions à ce niveau.

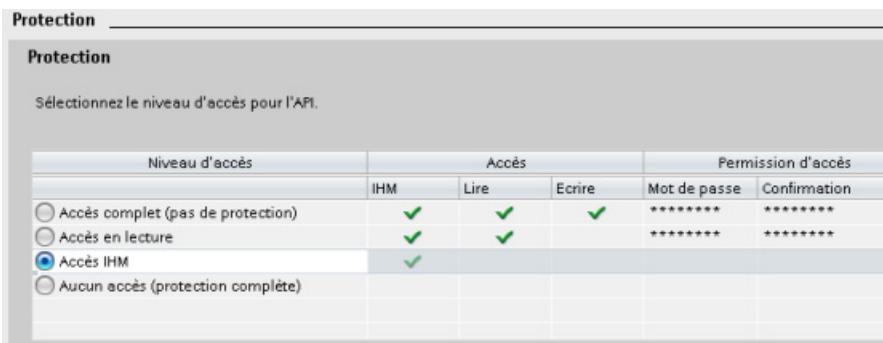
Les communications API-API (par le biais d'instructions de communication dans les blocs de code) ne sont pas réduites par le niveau de sécurité dans la CPU.

Tableau 5- 4 Niveaux de sécurité pour la CPU

Niveau de sécurité	Restrictions d'accès
Accès intégral (aucune protection)	Autorise un accès intégral sans protection par mot de passe.
Accès en lecture	Permet l'accès IHM et toutes les formes de communication API-API sans protection par mot de passe. Un mot de passe est nécessaire pour modifier la CPU (écriture) et pour changer le mode de la CPU (MARCHE/ARRET).
Accès IHM	Permet l'accès IHM et toutes les formes de communication API-API sans protection par mot de passe. Un mot de passe est nécessaire pour lire les données dans la CPU, modifier la CPU (écriture) et changer le mode de la CPU (MARCHE/ARRET).
Pas d'accès (protection intégrale)	N'autorise aucun accès sans protection par mot de passe. Un mot de passe est requis pour l'accès à une IHM, pour lire les données dans la CPU et pour modifier (écrire dans) la CPU.

Il y a distinction entre majuscules et minuscules dans le mot de passe. Procédez comme suit pour configurer le niveau de protection et les mots de passe :

1. Sélectionnez la CPU dans la "Configuration des appareils".
2. Dans la fenêtre d'inspection, sélectionnez l'onglet "Propriétés".
3. Sélectionnez la propriété "Protection" pour choisir le niveau de protection et entrer des mots de passe.



Quand vous téléchargez cette configuration dans la CPU, l'utilisateur a un accès IHM et peut avoir accès aux fonctions IHM sans mot de passe. Pour lire des données, l'utilisateur doit entrer le mot de passe configuré pour "l'Accès en lecture" ou le mot de passe pour l'"Accès intégral (aucune protection)". Pour écrire des données, l'utilisateur doit entrer le mot de passe configuré pour "l'Accès intégral (aucune protection)".

ATTENTION

Accès non autorisé à une CPU protégée

Les utilisateurs disposant de droits d'accès intégraux à la CPU ont des droits de lecture et d'écriture des variables API. Indépendamment du niveau d'accès pour la CPU, les utilisateurs de serveur Web peuvent avoir des droits de lecture et d'écriture de variables API. L'accès de personnes non autorisées à la CPU ou l'affectation de valeurs invalides aux variables API peut perturber le fonctionnement du processus, ce qui peut entraîner la mort, des blessures graves et/ou des dommages matériels importants.

Les utilisateurs non autorisés peuvent effectuer des modifications d'état de fonctionnement, des écritures sur les données API et des mises à jour de firmware. Siemens vous recommande d'observer les pratiques de sécurité suivantes :

- Niveaux d'accès à la CPU protégés par mot de passe et identifiants utilisateur de serveur Web (Page 264) avec des mots de passe forts. Les mots de passe forts contiennent au moins dix caractères, lettres, chiffres et caractères spéciaux mélangés, ne sont pas des mots du dictionnaire et ne sont pas des noms ou identifiants pouvant être déduits de vos données personnelles. Ne divuluez pas le mot de passe et changez-en fréquemment.
- N'autorisez l'accès au serveur Web qu'avec le protocole HTTPS.
- Ne pas étendre les droits minimaux par défaut de l'utilisateur "Tout le monde" du serveur Web.
- Effectuez des recherches d'erreur et des vérifications de plage sur vos variables dans la logique de programme, car les utilisateurs des pages Web ont la possibilité de donner des valeurs invalides aux variables API.

Mécanismes de liaison :

Pour avoir accès aux partenaires de connexion à distance avec les instructions PUT/GET, l'utilisateur doit également avoir une permission.

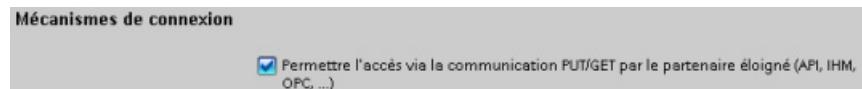
Par défaut, l'option "Permettre l'accès avec la communication PUT/GET" n'est pas activée. Dans ce cas, l'accès en lecture et en écriture aux données de la CPU n'est possible que pour des connexions de communication qui nécessitent une configuration ou une programmation pour la CPU locale et pour le partenaire de communication. L'accès par le biais des instructions BSEND/BRCV est possible, par exemple.

Les connexions pour lesquelles la CPU locale n'est qu'un serveur (ce qui veut dire qu'aucune configuration/programmation de la communication avec le partenaire de communication n'existe au niveau de la CPU locale), sont par conséquent impossibles pendant le fonctionnement de la CPU, par exemple :

- accès PUT/GET, FETCH/WRITE ou FTP par le biais de modules de communication
- accès PUT/GET à partir d'autres CPU S7
- accès IHM par le biais de la communication PUT/GET

Si vous voulez autoriser l'accès aux données de la CPU du côté client, autrement dit, si vous ne voulez pas restreindre les services de communication de la CPU, suivez les étapes suivantes :

1. configurez le niveau d'accès de protection pour qu'il soit n'importe quel niveau sauf "Aucun accès (protection intégrale)".
2. Cochez la case "Permettre l'accès avec la communication PUT/GET".



Quand vous téléchargez cette configuration sur la CPU, la CPU permet la communication PUT/GET à partir de partenaires distants

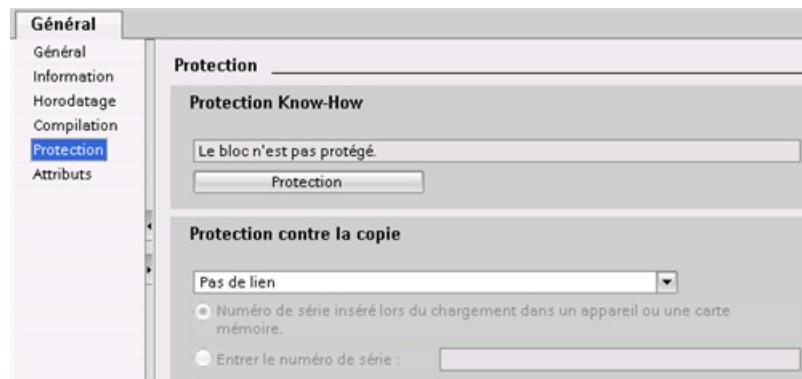
5.8.1 Protection du savoir-faire

Une protection "know-how" vous permet d'empêcher l'accès non autorisé à un ou plusieurs blocs de code (OB, FB, FC ou DB) dans votre programme. Vous créez un mot de passe pour limiter l'accès au bloc de code. La protection par mot de passe empêche de lire ou de modifier le bloc de code sans autorisation. Sans mot de passe, vous pouvez lire uniquement les informations suivantes sur le bloc de code :

- Titre, commentaire et propriétés du bloc
- Paramètres transmis (IN, OUT, IN_OUT, renvoi)
- Structure d'appel du programme
- Variables globales dans les références croisées (sans information sur l'endroit où elles sont utilisées) ; les variables locales sont masquées

Lorsque vous activez la protection "know-how" pour un bloc, le code à l'intérieur de ce bloc n'est accessible qu'après saisie du mot de passe.

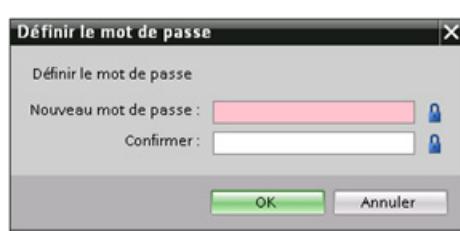
Utilisez la Task Card "Propriétés" du bloc de code pour configurer la protection "know-how" pour ce bloc. Après avoir ouvert le bloc de code, sélectionnez "Protection" dans les propriétés.



1. Dans les propriétés du bloc de code, cliquez sur le bouton "Protection" pour afficher la boîte de dialogue "Protection Know-How".
2. Cliquez sur le bouton "Définir" pour entrer le mot de passe.



Après avoir entré et confirmé le mot de passe, cliquez sur "OK".



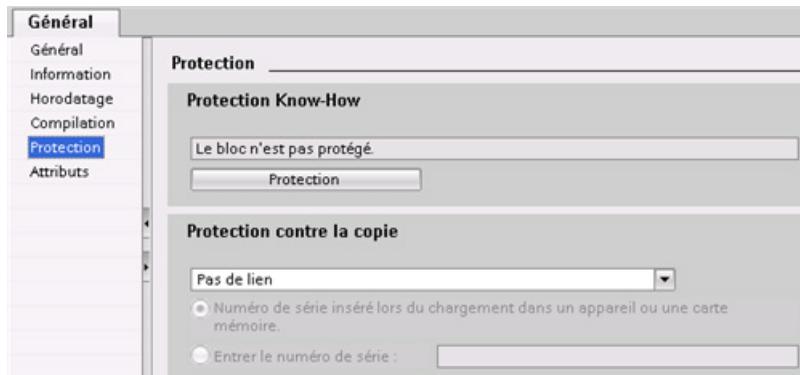
5.8.2 Protection contre la copie

Une fonction de sécurité supplémentaire vous permet de lier des blocs de programme à une utilisation avec une carte mémoire ou une CPU spécifique. Cette fonction est particulièrement utile pour protéger votre propriété intellectuelle. Lorsque vous liez un bloc de programme à un appareil spécifique, vous restreignez le programme ou le bloc de code pour une utilisation uniquement avec une carte mémoire ou une CPU spécifique. Cette fonction vous permet de distribuer un programme ou un bloc de code électroniquement (via Internet ou par email) ou en envoyant une cartouche mémoire. La protection contre la copie est disponible pour des OB (Page 97), des FB (Page 100) et des FC (Page 100). La CPU S7-1200 accepte trois types de protection de bloc :

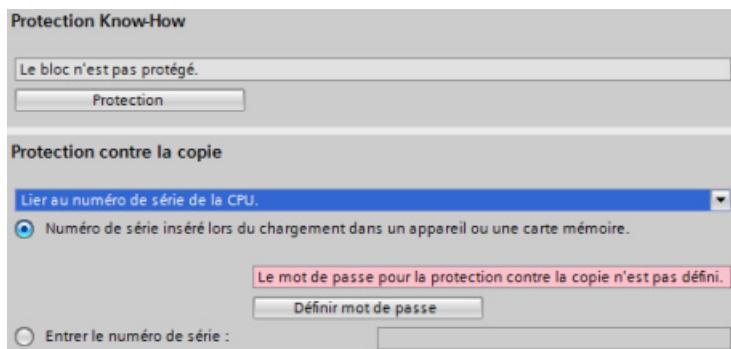
- Liaison avec le numéro de série d'une CPU
- Liaison avec le numéro de série d'une carte mémoire
- Liaison dynamique avec un mot de passe obligatoire

Utilisez la Task Card "Propriétés" du bloc de code pour lier le bloc à une CPU ou une carte mémoire spécifique.

1. Après avoir ouvert le bloc de code, sélectionnez "Protection".



2. Dans la liste déroulante sous la tâche "Protection contre la copie", sélectionnez le type de protection contre la copie que vous souhaitez utiliser.



3. Pour une liaison avec le numéro de série d'une CPU ou d'une carte mémoire, sélectionnez soit d'insérer le numéro de série lors du téléchargement, soit d'entrer le numéro de série de la carte mémoire ou de la CPU.

Remarque

Il y a distinction entre majuscules et minuscules dans le numéro de série.

Pour une liaison dynamique avec un mot de passe obligatoire, définissez le mot de passe que vous devez utiliser pour télécharger ou copier le bloc.

Lorsque vous téléchargez ultérieurement un bloc avec une liaison dynamique, vous devez entrer le mot de passe pour pouvoir télécharger le bloc. A noter que le mot de passe de protection contre la copie et le mot de passe de protection know-how (Page 92) sont deux mots de passe distincts.

Programmation facilitée

6.1

Conception aisée de votre programme

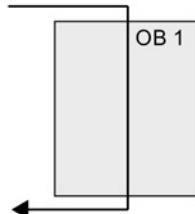
Lorsque vous créez un programme utilisateur pour les tâches d'automatisation, vous insérez les instructions pour le programme dans des blocs de code (OB, FB ou FC).

Choix du type de structure pour votre programme utilisateur

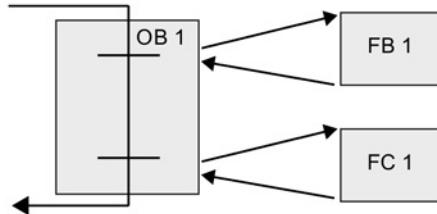
En fonction des besoins de votre application, vous pouvez choisir soit une structure linéaire soit une structure modulaire pour votre programme utilisateur.

- Un programme linéaire exécute toutes les instructions pour vos tâches d'automatisation séquentiellement les unes après les autres. En règle générale, le programme linéaire met toutes les opérations de programme dans un OB de cycle de programme (comme OB 1) pour l'exécution cyclique du programme.
- Un programme modulaire appelle des blocs de code spécifiques qui exécutent des tâches spécifiques. Pour créer une structure modulaire, vous divisez la tâche d'automatisation complexe en tâches subordonnées plus petites qui correspondent aux tâches fonctionnelles du processus. Chaque bloc de code fournit le segment de programme pour une tâche subordonnée. Vous structurez votre programme en appelant l'un des blocs de code à partir d'un autre bloc.

Structure linéaire :



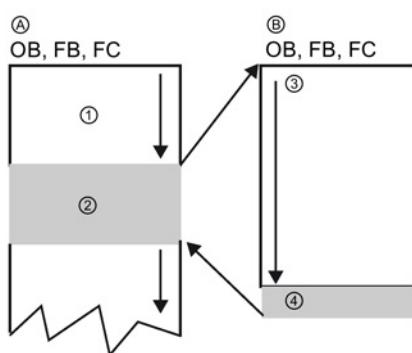
Structure modulaire :



En concevant des FB et des FC qui exécutent des tâches génériques, vous créez des blocs de code modulaires. Vous structurez ensuite votre programme utilisateur de sorte à ce que d'autres blocs de code appellent ces modules réutilisables. Le bloc appelant transmet des paramètres spécifiques de l'appareil au bloc appelé. Lorsqu'un bloc de code appelle un autre bloc de code, la CPU exécute le code du programme dans le bloc appelé. Une fois l'exécution du bloc appelé achevée, la CPU reprend l'exécution du bloc appelant. Le traitement se poursuit par l'exécution de l'instruction qui suit l'appel de bloc.

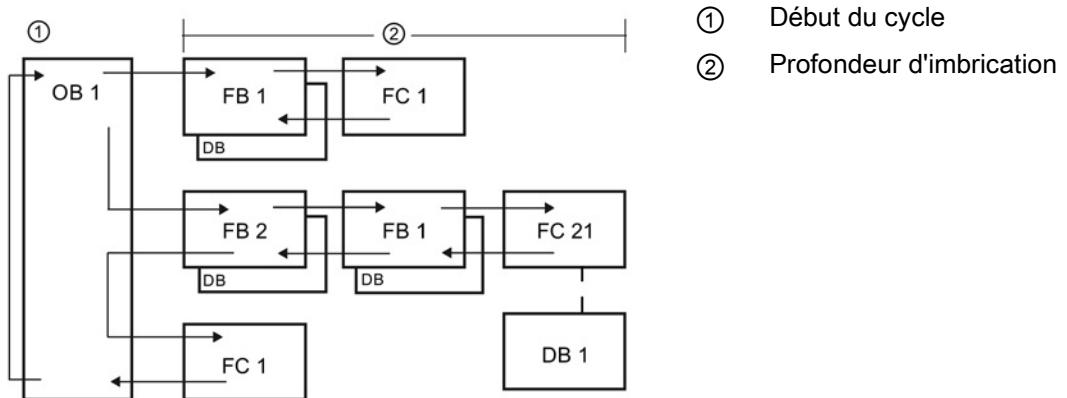
6.1 Conception aisée de votre programme

Vous pouvez également attribuer un OB à un événement d'alarme. Lorsque l'événement survient, la CPU exécute le code du programme dans l'OB associé. Une fois l'exécution de l'OB terminée, la CPU résume l'exécution au point dans le programme utilisateur où l'événement d'alarme est survenu, ce qui peut correspondre à n'importe quel point du cycle.



- A Bloc appelant (ou bloc interrompu)
- B FB ou BC appelé (ou OB d'alarme)
- ① Exécution du programme
- ② Opération (ou événement d'alarme) qui initialise l'exécution d'un autre bloc
- ③ Exécution du programme
- ④ Fin du bloc (retour au bloc appelant)

Vous pouvez imbriquer les appels de bloc pour obtenir une structure plus modulaire. Dans l'exemple suivant, la profondeur d'imbrication est 3 : de cycle de programme de l'OB plus 3 niveaux d'appels de blocs de code.



En créant des blocs de code génériques pouvant être réutilisés à l'intérieur du programme utilisateur, vous simplifiez la conception et l'implémentation du programme utilisateur.

- Vous pouvez créer des blocs de code réutilisables pour des tâches standard, telles que la commande d'une pompe ou d'un moteur. Vous pouvez également stocker ces blocs de code génériques dans une bibliothèque qui peut être utilisée par d'autres applications ou solutions.
- Lorsque vous structurez le programme utilisateur en composants modulaires liés à des tâches fonctionnelles, votre programme devient plus facile à comprendre et à gérer. Non seulement les composants modulaires vous aident à standardiser la conception du programme, mais ils rendent également l'actualisation ou la modification du code de programme plus rapide et plus facile.
- La création de composants modulaires simplifie le débogage de votre programme. En structurant le programme complet en tant qu'ensemble de segments de programme modulaires, vous pouvez tester la fonctionnalité de chaque bloc de code dès qu'il est élaboré.
- L'utilisation d'une conception modulaire rapportée à des tâches fonctionnelles spécifiques permet de diminuer le temps requis pour la mise en route de l'application.

6.1.1

Utilisation d'OB pour l'organisation de votre programme utilisateur

Les blocs d'organisation permettent de structurer votre programme. Ils servent d'interface entre le système d'exploitation et le programme utilisateur. Les OB sont déclenchés sur événement. La CPU exécute un OB à la suite d'un événement, tel qu'une alarme de diagnostic ou un intervalle de temps. Certains OB ont des événements déclencheurs et un comportement prédéfinis.

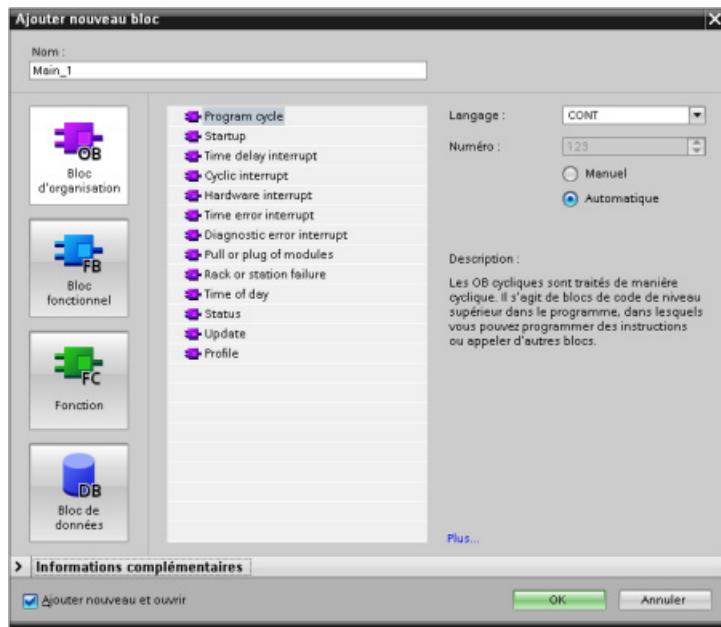
L'OB du cycle de programme contient votre programme principal. Vous pouvez inclure plus d'un OB de cycle de programme dans votre programme utilisateur. A l'état MARCHE, les OB de cycle de programme s'exécutent au niveau de priorité le plus bas et peuvent être interrompus par tous les autres types d'événements. L'OB de démarrage n'interrompt pas l'OB de cycle de programme, car la CPU exécute l'OB de démarrage avant de passer à l'état MARCHE.

Une fois le traitement des OB de cycle de programme achevé, la CPU réexécute immédiatement les OB de cycle de programme. Ce traitement cyclique est le type "normal" de traitement utilisé par les automates programmables. Pour de nombreuses applications, le programme utilisateur entier se trouve dans un OB de cycle de programme unique.

Vous pouvez créer d'autres OB qui exécuteront des fonctions spécifiques, telles que la gestion des alarmes et des erreurs ou l'exécution d'un code spécifique à intervalles spécifiques. Ces OB interrompent l'exécution des OB du cycle de programme.

6.1 Conception aisée de votre programme

Utilisez la boîte de dialogue "Ajouter nouveau bloc" pour créer de nouveaux OB dans votre programme utilisateur.



La gestion des alarmes se fait toujours sur événement. Lorsqu'un tel événement se produit, la CPU interrompt l'exécution du programme utilisateur et appelle l'OB qui a été configuré pour gérer cet événement. Une fois l'exécution de l'OB d'interruption achevée, la CPU reprend l'exécution du programme utilisateur à l'endroit où elle s'était interrompue.

La CPU détermine l'ordre de traitement des événements des alarmes par priorité. Vous pouvez affecter de multiples événements d'alarme à la même classe de priorité. Pour plus d'informations, reportez-vous aux rubriques blocs d'organisation (Page 58) et exécution du programme utilisateur (Page 57).

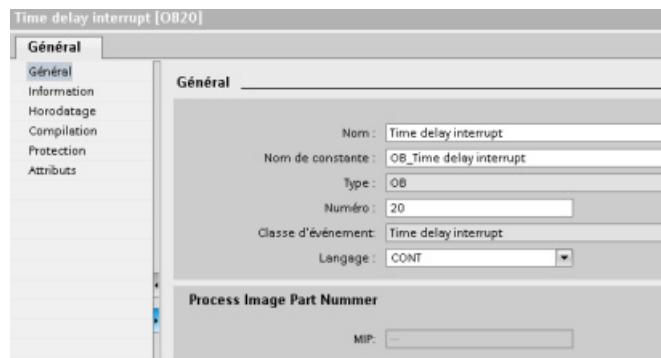
Création d'OB supplémentaires

Vous pouvez créer plusieurs OB pour votre programme utilisateur, et ce même pour les événements OB de cycle de programme et OB de démarrage. Utilisez la boîte de dialogue "Ajouter nouveau bloc" pour créer un OB et entrer le nom de votre OB.

Si vous créez plusieurs OB de cycle de programme pour votre programme utilisateur, la CPU exécute chaque OB de cycle de programme par ordre numérique, en commençant par l'OB de cycle de programme de plus petit numéro (l'OB 1, par exemple). Par exemple, une fois le premier OB de cycle de programme (tel l'OB 1) achevé, la CPU exécute l'OB de cycle de programme de numéro immédiatement supérieur.

Configuration des propriétés d'un OB

Vous pouvez modifier les propriétés d'un OB. Par exemple, vous pouvez configurer le nombre d'OB ou le langage de programmation.



Remarque

Notez que vous pouvez affecter à un OB un numéro de référence de la mémoire image qui correspond à PIP0, PIP1, PIP2, PIP3 ou PIP4. Si vous entrez un nombre pour le numéro de référence de la mémoire image, la CPU crée cette mémoire image partielle. Reportez-vous à la rubrique "Exécution du programme utilisateur (Page 57)" pour une explication des mémoires images partielles.

6.1.2

Programmation aisée des tâches modulaires avec les FB et FC

Une fonction (FC) est comme un sous-programme. Une FC est un bloc de code qui exécute typiquement une opération spécifique sur un ensemble de valeurs d'entrée. La FC stocke les résultats de cette opération dans des adresses de mémoire. Utilisez des FC pour les tâches suivantes :

- Exécuter des opérations standard et réutilisables, par exemple pour des calculs mathématiques
- Exécuter des tâches fonctionnelles, par exemple pour des commandes individuelles à l'aide d'opérations logiques sur bits

Une FC peut également être appelée plusieurs fois en différents points d'un programme. Cette réutilisation simplifie la programmation de tâches revenant souvent.

Contrairement à un FB, une FC ne comporte pas de DB d'instance associé. La FC se sert de sa mémoire temporaire (L) pour les données nécessaires au calcul de l'opération. Les données temporaires ne sont pas sauvegardées. Pour stocker les données en vue de leur utilisation une fois l'exécution de la FC terminée, affectez la valeur de sortie à une adresse de mémoire globale, telle qu'un mémento M ou un DB global.

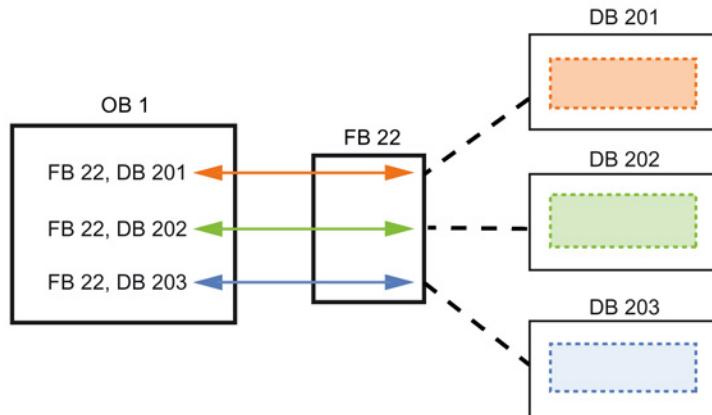
Un bloc fonctionnel (FB) est comme un sous-programme avec une mémoire. Un FB est un bloc de code dont les appels peuvent être programmés au moyen de paramètres de bloc. Le FB stocke les paramètres d'entrée (IN), de sortie (OUT) et d'entrée/sortie (IN_OUT) dans une mémoire variable placée dans un bloc de données (DB) ou dans un DB d'instance. Le DB d'instance fournit un bloc de mémoire qui est associé à cette instance (ou appel) du FB et qui contient les données une fois le FB achevé.

Vous utiliserez typiquement un FB pour commander l'exécution de tâches ou le fonctionnement d'appareils qui durent plus d'un cycle. Pour stocker les paramètres de fonctionnement afin qu'ils soient rapidement accessibles d'un cycle au suivant, chaque FB dans votre programme utilisateur comporte un ou plusieurs DB d'instance. Lorsque vous appelez un FB, vous ouvrez aussi un DB d'instance qui mémorise les valeurs des paramètres du bloc et les données locales statiques pour cet appel ou "instance" du FB. Ces valeurs sont stockées dans le DB d'instance une fois le FB terminé.

Vous pouvez affecter des valeurs initiales aux paramètres dans l'interface du FB. Ces valeurs sont transférées dans le DB d'instance associé. Si vous n'affectez pas de valeurs aux paramètres, les valeurs actuellement stockées dans le DB d'instance seront utilisées. Dans certains cas, vous devez affecter des valeurs aux paramètres.

Vous pouvez associer différents DB d'instance à différents appels du FB. Les DB d'instance vous permettent d'utiliser un FB générique pour commander plusieurs appareils. Vous structurez votre programme en insérant dans un bloc de code l'appel d'un FB et d'un DB d'instance. La CPU exécute alors le code dans ce FB et stocke les paramètres du bloc et les données locales statiques dans le DB d'instance. Une fois l'exécution du FB achevée, la CPU revient au bloc de code qui a appelé le FB. Le DB d'instance conserve les valeurs pour cette instance du FB. En concevant le FB pour des tâches de commande génériques, vous pouvez le réutiliser pour plusieurs appareils en sélectionnant des DB d'instance différents pour différents appels du FB.

La figure suivante montre un OB qui appelle un FB trois fois, avec un bloc de données différent à chaque appel. Cette structure permet à un FB générique de commander plusieurs appareils similaires, tels des moteurs, en affectant un bloc de données d'instance différent à chaque appel pour les différents appareils.



Chaque DB d'instance mémorise les données (par exemple, la vitesse, le temps d'accélération et le temps de fonctionnement total) pour un appareil distinct. Dans cet exemple, le FB 22 commande trois appareils distincts, le DB 201 contenant les données de fonctionnement du premier appareil, le DB 202 celles du deuxième appareil et le DB 203 celles du troisième appareil.

6.1.3

Stockage aisé des données de programme dans des blocs de données

Vous créez des blocs de données (DB) dans votre programme utilisateur pour sauvegarder les données des blocs de code. Tous les blocs de code dans le programme utilisateur peuvent accéder aux données d'un DB global. En revanche, un DB d'instance contient des données pour un bloc fonctionnel spécifique (FB).

Votre programme utilisateur peut sauvegarder les données dans les zones de mémoire spécialisées de la CPU, à savoir les entrées (I), les sorties (Q) et les mémentos (M). En outre, vous pouvez utiliser un bloc de données (DB) pour accéder rapidement aux données stockées dans le programme lui-même.

Les données sauvegardées dans un DB ne sont pas effacées à la fermeture du bloc de données ni à la fin de l'exécution du bloc de code associé. Il existe deux types de DB :

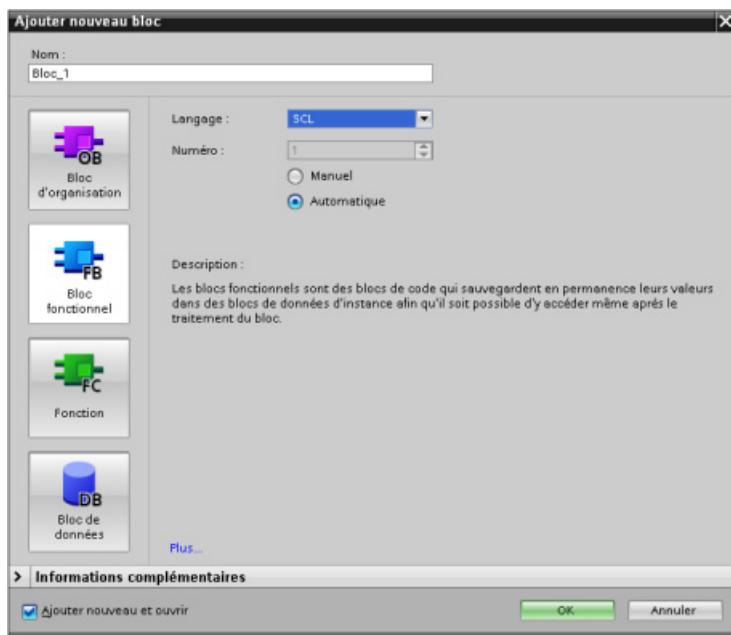
- Un DB global contient des données pour les blocs de code dans votre programme. Tous les OB, FB et FC peuvent accéder aux données dans un DB global.
- Un DB d'instance contient les données d'un FB spécifique. La structure des données dans un DB d'instance reflète les paramètres (Input, Output et InOut) et les données statiques du FB. La mémoire Temp pour le FB n'est pas stockée dans le DB d'instance.

Bien que le DB d'instance reflète les données d'un FB spécifique, tout bloc de code peut accéder aux données à l'intérieur d'un DB d'instance.

6.1.4 Crédation d'un nouveau bloc de code

Pour ajouter un bloc de code au programme, suivez les étapes ci-dessous :

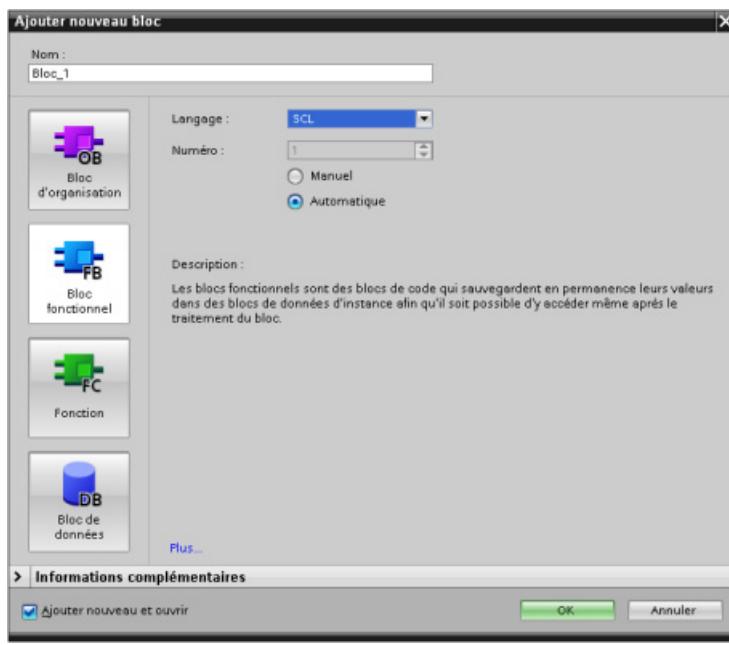
1. Ouvrez le dossier "Blocs de programme".
2. Double-cliquez sur "Ajouter nouveau bloc".
3. Dans la boîte de dialogue "Ajouter un bloc", cliquez sur le type de bloc à ajouter. Par exemple, cliquez sur l'icône "Fonction (FC)" pour ajouter une FC.
4. Sélectionnez le langage de programmation pour le bloc de code dans le menu déroulant.



5. Cliquez sur "OK" pour ajouter le bloc au projet.

Lorsque vous sélectionnez l'option (par défaut) "Ajouter nouveau et ouvrir", STEP 7 ouvre le bloc qui vient d'être créé, dans l'éditeur.

6.1.5 Crédation de blocs de code réutilisables



Utilisez la boîte de dialogue "Ajouter nouveau bloc" sous "Blocs de programme" dans le navigateur du projet pour créer des OB, des FB, des FC et des DB globaux.

Lors de la création d'un bloc de code, vous sélectionnez le langage de programmation pour ce bloc. Vous ne sélectionnez pas de langage pour les DB qui servent uniquement au stockage de données.

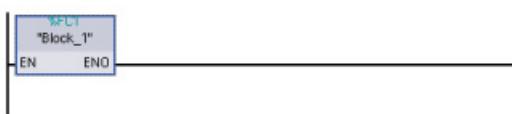
Lorsque vous cochez la case (par défaut) "Ajouter nouveau et ouvrir", le bloc de code s'ouvre dans la vue du projet.

Vous pouvez stocker des objets que vous voulez réutiliser dans des bibliothèques. Pour chaque projet, il y a une bibliothèque de projets qui est connectée au projet. Outre la bibliothèque de projet, vous pouvez créer n'importe quel nombre de bibliothèques internes qui peuvent être utilisées sur plusieurs projets. Les bibliothèques étant compatibles les unes avec les autres, des éléments de bibliothèque peuvent être copiés et retirés d'une bibliothèque sur une autre.

Les bibliothèques sont utilisées, par exemple, pour créer des modèles pour des blocs que vous collez d'abord dans la bibliothèque de projet, puis que vous développez davantage à cet endroit. Enfin, vous copiez les blocs de la bibliothèque du projet sur une bibliothèque globale. Vous rendez la bibliothèque globale disponible pour d'autres collègues qui travaillent sur votre projet. Ils utilisent les blocs et les adaptent en outre aux besoins individuels, si nécessaire.

Pour avoir des précisions sur les mises en œuvre de bibliothèques, reportez-vous aux rubriques de bibliothèques Aide en ligne sur STEP 7.

6.1.6 Appel d'un bloc de code à partir d'un autre bloc de code



Vous pouvez aisément appeler un bloc de code quelconque (OB, FB ou FC) dans votre programme utilisateur en l'appelant à partir d'un FB ou d'une FC dans votre CPU.

1. Ouvrez le bloc de code à partir duquel appeler l'autre bloc.
2. Sélectionnez le bloc de code à appeler dans l'arborescence du projet.
3. Amenez le bloc sur le réseau sélectionné afin de créer une instruction Call sur le bloc de code.

Remarque

Votre programme utilisateur ne peut pas appeler d'OB, car ceux-ci sont pilotés par événement (Page 58). La CPU démarre l'exécution de l'OB en réponse à la réception d'un événement.

6.2

Langages de programmation faciles à utiliser

STEP 7 fournit les langages de programmation standard suivants pour le S7-1200 :

- CONT (schéma à contacts) est un langage de programmation graphique. Sa représentation se base sur des schémas (Page 105) de circuit.
- LOG (logigramme) est un langage de programmation se fondant sur les symboles logiques graphiques utilisés en algèbre (Page 106) booléenne.
- SCL (Structured Control Language) est un langage de programmation littéral évolué (Page 107).

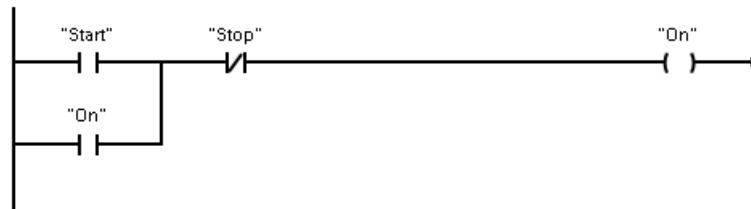
Lorsque vous créez un bloc de code, vous sélectionnez le langage de programmation à utiliser par ce bloc.

Votre programme utilisateur peut utiliser des blocs de code créés dans n'importe lequel des langages de programmation disponibles.

6.2.1

Schéma à contacts (CONT)

Les éléments d'un schéma de circuit, contacts à ouverture et à fermeture et bobines par exemple, sont reliés pour former des réseaux.



Pour créer la logique pour des opérations complexes, vous pouvez insérer des branches formant des circuits parallèles. Les branches parallèles s'ouvrent vers le bas ou se connectent directement à la barre conductrice. Vous terminez les branches vers le haut.

CONT fournit des instructions sous forme de boîtes pour des fonctions variées, telles que les fonctions mathématiques, de temporisation, de comptage et de transfert.

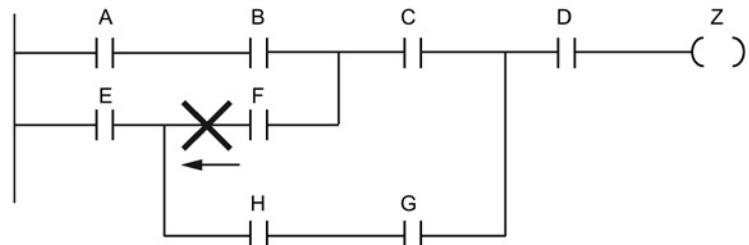
STEP 7 ne limite pas le nombre d'instructions (lignes et colonnes) dans un réseau CONT.

Remarque

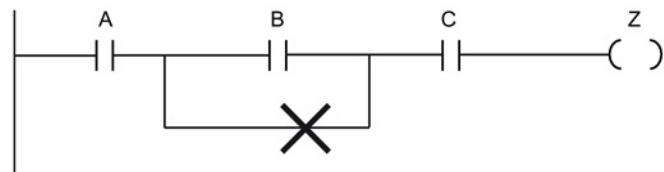
Chaque réseau CONT doit se terminer par une bobine ou une boîte d'instruction.

Vous devez tenir compte des règles suivantes lors de la création d'un réseau CONT :

- Vous ne pouvez pas créer de branche qui entraînerait un flux de courant en sens inverse.

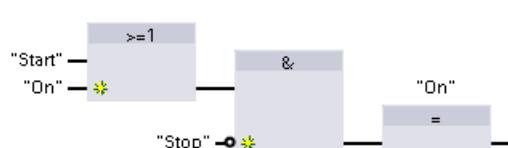


- Vous ne pouvez pas créer de branche qui provoquerait un court-circuit.



6.2.2 Logigramme (LOG)

Comme CONT, LOG est un langage de programmation graphique. La représentation de la logique repose sur les symboles logiques graphiques utilisés en algèbre booléenne.



Pour créer la logique pour des opérations complexes, insérez des branches parallèles entre les boîtes.

Les fonctions mathématiques et autres fonctions complexes peuvent être représentées directement avec des boîtes logiques.

STEP 7 ne limite pas le nombre d'instructions (lignes et colonnes) dans un réseau LOG.

6.2.3 Vue d'ensemble du SCL

SCL (Structured Control Language) est un langage de programmation PASCAL évolué pour les CPU SIMATIC S7. Le SCL prend en charge la structure de bloc de STEP 7. Vous pouvez également inclure des blocs de programme écrits dans le SCL avec des blocs de programme écrits dans CONT et LOG.

Les opérations SCL utilisent des opérateurs de programmation standards, comme par exemple pour l'attribution (:=), les fonctions mathématiques (+ pour l'addition, - pour la soustraction, * pour la multiplication et / pour la division). SCL utilise des opérations de contrôle de programme PASCAL standard, telles que IF-THEN-ELSE, CASE, REPEAT-UNTIL, GOTO et RETURN. Vous pouvez utiliser toute référence PASCAL aux éléments de syntaxe du langage de programmation SCL. Parmi les autres opérations pour SCL, comme les temporisations et les compteurs, un grand nombre correspond aux opérations CONT et LOG.

Parce que SCL, comme PASCAL, offre des structures de traitement conditionnel, de mise en boucle et de contrôle d'emboîtement, vous pouvez exécuter des algorithmes complexes dans SCL plus facilement que dans CONT ou LOG.

Les exemples suivants montrent différentes expressions pour différentes utilisations :

<code>"C" := #A+#B;</code>	Attribue deux variables locales à une variable
<code>"Data_block_1".Tag := #A;</code>	Affectation à une variable de bloc de données
<code>IF #A > #B THEN "C" := #A;</code>	Condition pour l'énoncé IF-THEN
<code>"C" := SQRT (SQR (#A) + SQR (#B));</code>	Paramètres pour l'opération SQRT

Comme un langage de programmation de haut niveau, le SCL utilise des énoncés standards pour les tâches basiques :

- Énoncé d'attribution :=
- Fonctions mathématiques : +, -, *, et /
- Adressage de variables globales : "<nom variable>" (nom de la variable ou du bloc de données entre guillemets)
- Adressage de variables locales : #<nom variable> (nom de la variable précédé du symbole "#")
- Adressage absolu : %<absolute address>, par exemple %I0.0 ou %MW10

Les opérateurs arithmétiques peuvent traiter divers types de données numériques. Le type de données du résultat est déterminé par le type de données des opérandes les plus significatives. Par exemple, une multiplication qui utilise une opérande INT et une opérande REAL donne une valeur REAL pour le résultat.

6.2.4 Éditeur de programme SCL

Vous pouvez spécifier, lors de la création de n'importe quel type de bloc (OB, FB ou FC), que ce bloc utilise le langage de programmation SCL. STEP 7 fournit un éditeur de programmes SCL qui inclut les éléments suivants :

- Section d'interface pour définir les paramètres du bloc de code
- Section de code pour le code de programme
- Arborescence d'instructions contenant les instructions SCL prises en charge par la CPU

Vous entrez le code SCL pour votre instruction directement dans la section de code.

L'éditeur comprend des boutons pour les structures de code usuelles et les commentaires. Pour des instructions plus complexes, faites simplement glisser les instructions SCL depuis l'arborescence d'instructions et déposez-les dans votre programme. Vous pouvez également utiliser un éditeur de texte quelconque pour créer un programme SCL et importer ensuite ce fichier dans STEP 7.

Function_1			
	Nom	Type de données	Commentaire
1	Input		
2	StartStopSwitch	Bool	
3	Output		
4	RunYesNo	Bool	
5	InOut		
6	<Add new>		
7	Temp		
8	<Add new>		
9	Constant		
10	<Add new>		
11	Return		
12	Function_1	Void	


```

IF... CASE... FOR... WHILE...
OF... TO DO... DO... (*...*)

1 IF condition THEN
2     // Statement section IF
3     ;
4 END_IF;

```

Vous pouvez déclarer les types de paramètres suivants dans la section d'interface du bloc de code SCL :

- Input, Output, InOut et Ret_Val. Ces paramètres définissent les variables d'entrée, les variables de sortie et la valeur en retour pour le bloc de code. Le nom de variable que vous entrez ici est utilisé localement lors de l'exécution du bloc de code. En règle générale, vous n'utilisez pas le nom de variable globale de la table de variables.
- Static (FB uniquement ; la figure ci-dessus concerne une FC) : Le bloc de code utilise des variables statiques pour stocker des résultats intermédiaires statiques dans le bloc de données d'instance. Le bloc conserve les données statiques jusqu'à leur écrasement, qui peut avoir lieu après plusieurs cycles. Les noms des blocs que ce bloc appelle comme multi-instance sont également stockés dans les données locales statiques.
- Temp : ces paramètres sont les variables temporaires utilisées lors de l'exécution du bloc de code.
- Constant: il s'agit de valeurs de constantes nommés pour votre bloc de code.

Si vous appelez le bloc de code SCL depuis un autre bloc de code, les paramètres du bloc de code SCL apparaissent comme entrées ou sorties.



Dans cet exemple, les variables pour "Start" et "On" (depuis la table de variables du projet) correspondent à "StartStopSwitch" et "RunYesNo" dans la table de déclaration du programme SCL.

6.3 Les opérations puissantes facilitent la programmation

6.3.1 Mise à disposition des instructions de base dont vous avez besoin

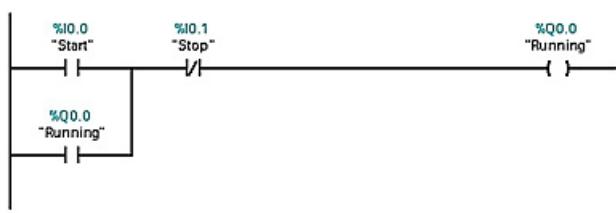
La CPU S7-1200 prend en charge de nombreuses instructions. Elles sont disponibles dans l'arborescence d'instructions de STEP 7 dans les groupes suivants :

- Instructions de base
- Instructions avancées
- Technologie
- Instruction de communication

Vous pouvez trouver une synthèse complète de toutes les instructions dans le *Manuel système de l'automate programmable S7-1200*. Ce manuel décrit de nombreuses instructions communes.

Opérations combinatoires sur bits

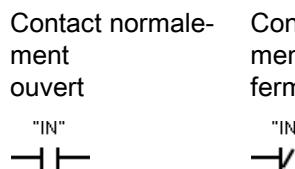
Les contacts et les bobines constituent la base de la logique sur bits. Les contacts servent à lire l'état d'un bit, alors que les bobine servent à écrire l'état de l'opération dans un bit.



Les contacts testent l'état binaire du bit, le résultat se traduisant par un "flux de courant" s'il est positif (1) ou une "absence de flux de courant" s'il est négatif (0).

L'état de la bobine reflète l'état de la logique précédente.

Si vous utilisez une bobine avec la même adresse dans plusieurs emplacements du programme, le résultat du dernier calcul dans le programme utilisateur détermine le statut de la valeur écrite dans la sortie physique lors de la mise à jour des sorties.

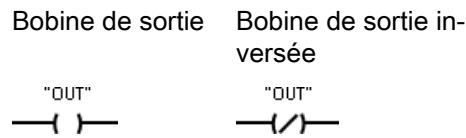


Le contact à fermeture est fermé (activé) lorsque la valeur du bit affecté est égale à 1.
Le contact à ouverture est fermé (activé) lorsque la valeur du bit affecté est égale à 0.

La structure de base d'une opération logique sur bits est soit une logique AND, soit une logique OR. Des contacts connectés en série créent des réseaux de logique ET. Des contacts connectés en parallèle créent des réseaux de logique OU.

Vous pouvez connecter des contacts à d'autres contacts et créer votre propre logique combinatoire. Si le bit d'entrée que vous indiquez utilise l'identificateur de mémoire I (entrée) ou Q (sortie), la valeur du bit est lue dans la mémoire image du processus. Les signaux du contact physique dans votre processus de régulation sont câblés aux bornes d'entrée de l'automate. La CPU lit les signaux d'entrée câblés et actualise les valeurs d'état correspondantes dans la mémoire image des entrées.

Vous pouvez effectuer une lecture directe d'une entrée physique en indiquant ":P" après la variable d'une entrée (p. ex. "Démarrage_Moteur:P" ou "I3.4:P"). En cas de lecture directe, les valeurs de données binaires sont lues directement dans l'entrée physique et non dans la mémoire image. Une lecture directe n'actualise pas la mémoire image.



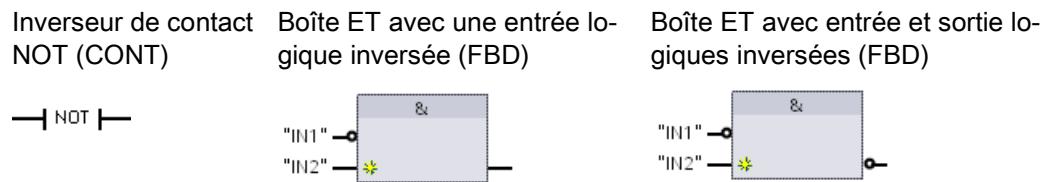
Notez les résultats de sortie suivants pour le débit de puissance par des bobines de sortie et des bobines de sortie inversées :

- S'il y a flux de courant à travers une bobine de sortie, le bit de sortie est mis à 1.
- S'il n'y a pas de flux de courant à travers une bobine de sortie, le bit de la bobine de sortie est mis à 0.
- S'il y a flux de courant à travers une bobine de sortie inversée, le bit de sortie est mis à 0.
- S'il n'y a pas de flux de courant à travers une bobine de sortie inversée, le bit de sortie est mis à 1.

L'instruction Bobine de sortie écrit une valeur pour un bit de sortie. Si le bit de sortie que vous indiquez utilise l'identificateur de mémoire Q, la CPU met le bit de sortie dans la mémoire image du processus à 1 ou à 0 conformément à l'état de flux de courant. Les signaux de sorties pour vos actionneurs de contrôle sont raccordés aux bornes de sorties sur l'API. A l'état MARCHE, la CPU lit les signaux des entrées, traite les états d'entrée en fonction de la logique de votre programme, puis réagit en entrant de nouvelles valeurs d'état de sortie dans la mémoire image du processus. Après chaque cycle d'exécution du programme, la CPU transfère le nouvel état des sorties mémorisé dans la mémoire image aux bornes de sortie câblées.

Vous pouvez effectuer une écriture directe d'une sortie physique en indiquant ":P" après la variable d'une sortie (p. ex. "Moteur_Marche:P" ou "Q3.4:P"). En cas d'écriture directe, les valeurs de données binaires sont écrites directement dans la mémoire image des sorties et directement dans la sortie physique.

Les bobines peuvent être utilisées ailleurs qu'à la fin d'un réseau. Vous pouvez insérer une bobine au milieu d'un barreau du réseau CONT, entre des contacts ou d'autres instructions.

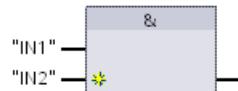


Le contact NOT en CONT inverse l'état logique de l'entrée de flux de courant.

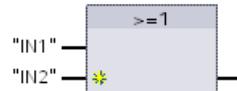
- S'il n'y a pas de flux de courant entrant dans le contact NOT, il y a un flux de courant sortant.
- S'il y a un flux de courant entrant dans le contact NOT, il n'y a pas de flux de courant sortant.

En programmation LOG, vous pouvez faire glisser l'outil "Inverser RLO" de la barre d'outils "Favoris" ou de l'arborescence d'instructions vers une entrée ou une sortie afin de créer un inverseur logique sur ce connecteur de boîte.

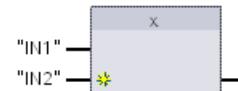
Boîte ET (FBD)



Boîte OU (FBD)



Boîte OU EXCLUSIF (FBD)



- Toutes les entrées d'une boîte ET doivent être vraies pour que la sortie soit vraie.
- Il suffit qu'une entrée d'une boîte OU soit vraie pour que la sortie soit vraie.
- Un nombre impair d'entrées d'une boîte OU EXCLUSIF doivent être vraies pour que la sortie soit vraie.

En programmation FBD, les réseaux de contacts CONT sont représentés par des réseaux de boîtes ET (&), OU (≥ 1) et OU EXCLUSIF (x) dans lesquels vous pouvez indiquer des valeurs binaires pour les entrées et sorties des boîtes. Vous pouvez aussi établir des connexions à d'autres boîtes logiques et créer vos propres combinaisons logiques. Une fois que vous avez placé la boîte dans votre réseau, vous pouvez faire glisser l'outil "Insérer entrée" de la barre d'outils "Favoris" ou de l'arborescence d'instructions vers le côté d'entrée de la boîte afin d'ajouter des entrées. Vous pouvez aussi cliquer avec le bouton droit de la souris sur le connecteur d'entrée de la boîte et sélectionner "Insérer entrée".

Les entrées et la sortie de la boîte peuvent être connectées à une autre boîte logique ou vous pouvez entrer une adresse binaire ou un mnémonique de bit pour une entrée non connectée. Lors de l'exécution de la boîte d'instruction, les états des entrées en cours sont appliqués à la logique binaire et, si elle est vraie, la sortie de la boîte sera vraie.

6.3.2 Comparaison et transfert

La Comparaison réalise une comparaison de deux valeurs avec le même type de données.

Tableau 6- 1 Opérations de comparaison

Instruction	SCL	Description
CONT : "IN1" == Byte "IN2"	<code>out := in1 = in2;</code> <code>out := in1 <> in2;</code> <code>out := in1 >= in2;</code> <code>out := in1 <= in2;</code> <code>out := in1 > in2;</code> <code>out := in1 < in2;</code>	<ul style="list-style-type: none"> • Egal à (==) : la comparaison est vraie si IN1 est égal à IN2 • Différent (<>) : la comparaison est vraie si IN1 n'est pas égal à IN2
LOG : "IN1" — IN1 "IN2" — IN2	<code>out := in1 < in2;</code>	<ul style="list-style-type: none"> • Supérieur ou égal à (>=) : la comparaison est vraie si IN1 est supérieur ou égal à IN2 • Inférieur ou égal à (<=) : la comparaison est vraie si IN1 est inférieur ou égal à IN2 • Supérieur à (>) : la comparaison est vraie si IN1 est supérieur à IN2 • Inférieur à (<) : la comparaison est vraie si IN1 est inférieur à IN2

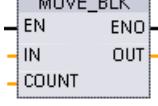
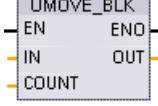
¹ pour CONT et LOG : le contact est activé (CONT) ou la sortie de boîte est VRAIE (LOG) si la comparaison est VRAIE,

Pour plus d'opérations de comparaison, référez-vous au *Manuel système de l'automate programmable S71200*.

Le Transfert copie les éléments de données vers une nouvelle adresse de mémoire et peut convertir un type de données en un autre. Les données source ne sont pas modifiées par le processus de transfert.

- MOVE copie un élément de données stocké à une adresse spécifiée vers une nouvelle adresse. Pour ajouter une autre sortie, cliquez sur l'icône à côté du paramètre OUT1.
- MOVE_BLK (transfert interruptible) et UMOVE_BLK (transfert sans coupure) copient un bloc d'éléments de données vers une nouvelle adresse. Les instructions MOVE_BLK et UMOVE_BLK ont un paramètre COUNT supplémentaire. Le COUNT spécifie combien d'éléments de données sont copiés. Le nombre d'octets par élément copié dépend du type de données attribué aux noms de variable de paramètres IN et OUT dans la table de variable API.

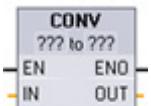
Tableau 6- 2 Instructions MOVE, MOVE_BLK et UMOVE_BLK

CONT/LOG	SCL	Description
	<code>out1 := in;</code>	Copie un élément de données stocké à une adresse spécifiée vers une ou plusieurs nouvelle(s) adresse(s). Pour ajouter une autre sortie dans CONT ou LOG, cliquez sur l'icone près du paramètre de sortie. Pour SCL, utilisez des énoncés d'attribution multiples. Vous pouvez également utiliser l'une des constructions en boucle.
	<code>MOVE_BLK(in:=_variant_in, count:=_uint_in, out=>_variant_out);</code>	Transfert interruptible qui copie un bloc d'éléments de données vers une nouvelle adresse.
	<code>UMOVE_BLK(in:=_variant_in, count:=_uint_in out=>_variant_out);</code>	Transfert sans coupure qui copie un bloc d'éléments de données vers une nouvelle adresse.

Pour plus d'opérations de transfert, référez-vous au *Manuel système de S71200*.

6.3.3 Conversion

Tableau 6- 3 Conversion

CONT/LOG	SCL	Description
	<code>out := <data type in>_TO_<data type out>(in);</code>	Convertit un élément de données à partir d'un type de données vers un autre type de données.

- 1 pour CONT et LOG : Cliquez sous le nom de boîte et sélectionner les types de données à partir du menu déroulant. Une fois que vous avez sélectionné le type de données (convertir de), une liste des conversions possibles s'affiche dans la liste déroulante (convertir en).
- 2 Pour SCL : Construisez l'instruction de conversion en identifiant le type de données pour le paramètre d'entrée et le paramètre de sortie. Par exemple, DWORD_TO_REAL convertit une valeur double mot en une valeur réelle.

Tableau 6- 4 Instructions Round et Truncate

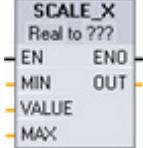
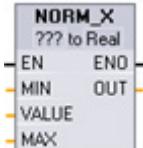
CONT/LOG	SCL	Description
	<code>out := ROUND (in);</code>	<p>Convertit un nombre réel (Real or LReal) en un nombre entier. L'instruction arrondit le nombre réel à la valeur entière la plus proche (IEEE, arrondir au plus proche). Si le nombre est exactement la moitié de l'espace entre deux entiers (par exemple 10,5), alors le nombre est arrondi au nombre entier. Par exemple :</p> <ul style="list-style-type: none"> • ROUND (10.5) = 10 • ROUND (11.5) = 12 <p>Pour LAD/FBD, vous cliquez sur "???" dans la boîte à instructions pour sélectionner le type de données pour la sortie, par exemple, "DInt". Pour SCL, le type de données de sortie par défaut est DINT. Pour arrondir à un autre type de données de sortie, entrez le nom de l'instruction avec le nom explicite du type de données, par exemple, ROUND_REAL or ROUND_LREAL.</p>
	<code>out := TRUNC (in);</code>	Convertit un nombre réel (Real or LReal) en un nombre entier. La partie fractionnaire du nombre réel est tronquée à zéro (IEEE, arrondir à zéro).

6.3 Les opérations puissantes facilitent la programmation

Tableau 6- 5 Instructions plafond (CEIL) et Floor

CONT/LOG	SCL	Description
	<code>out := CEIL(in);</code>	Convertit un nombre réel (Real ou LReal) au nombre entier le plus proche supérieur ou égal au nombre réel sélectionné (IEEE "arrondi à +infini").
	<code>out := FLOOR(in);</code>	Convertit un nombre réel (Real ou LReal) au nombre entier le plus proche inférieur ou égal au nombre réel sélectionné (IEEE "arrondi à -infini").

Tableau 6- 6 Instructions SCALE_X et NORM_X

CONT/LOG	SCL	Description
	<code>out := SCALE_X(</code> <code>min:=_in_,</code> <code>value:=_in_,</code> <code>max:=_in_);</code>	Graduer la VALEUR de paramètre réel normalisée lorsque $(0,0 \leq VALEUR \leq 1,0)$ dans le type de données et la portée de valeur spécifiés par les paramètres MIN et MAX : $OUT = VALUE \cdot (MAX - MIN) + MIN$
	<code>out := NORM_X(</code> <code>min:=_in_,</code> <code>value:=_in_,</code> <code>max:=_in_);</code>	Normalise la VALEUR de paramètre dans la portée de valeur spécifiée par les paramètres MIN et MAX : $OUT = (VALUE - MIN) / (MAX - MIN)$, où $(0,0 \leq OUT \leq 1,0)$

¹ Équivalent SCL : `out := value * (max-min) + min;`² Équivalent SCL : `out := (value-min) / (max-min);`

6.3.4 Les maths simplifiés avec l'opération Calculer

Tableau 6- 7 Instruction CALCULATE

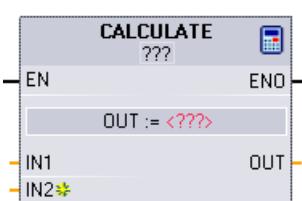
CONT/LOG	SCL	Description
	<p>Utilisez les expressions mathématiques SCL standard pour créer l'équation.</p>	<p>L'instruction CALCULATE permet de créer une fonction mathématique qui opère sur les entrées (IN1, IN2, ... INn) et fournit le résultat dans OUT, conformément à l'équation que vous définissez.</p> <ul style="list-style-type: none"> • Sélectionnez d'abord un type de données. Toutes les entrées et la sortie doivent avoir le même type de données. • Pour ajouter une autre entrée, cliquez sur l'icône à côté de la dernière entrée.

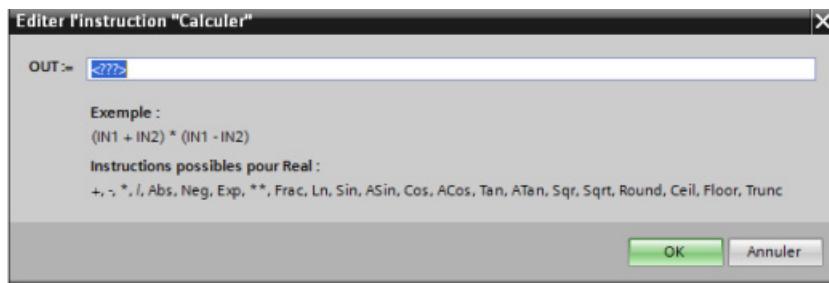
Tableau 6- 8 Types de données pour les paramètres

Paramètre	Type de données ¹
IN1, IN2, ...INn	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord

- ¹ Les paramètres IN et OUT doivent avoir le même type de données (avec conversion implicite des paramètres d'entrée). Par exemple, une valeur SINT pour une entrée sera convertie en valeur INT ou REAL si OUT est un entier (INT) ou un réel (REAL).

Cliquez sur l'icône du calculateur pour ouvrir la boîte de dialogue et définir votre fonction mathématique. Vous entrez votre équation sous forme d'entrées (telles que IN1 et IN2) et d'opérations. Lorsque vous cliquez sur "OK" pour sauvegarder la fonction, le dialogue crée automatiquement les entrées pour l'instruction CALCULATE.

La boîte de dialogue représente un exemple et une liste d'opérations possibles que vous pouvez inclure en fonction du type de données du paramètre OUT :



Remarque

Vous devez également créer une entrée pour chaque constante dans votre fonction. La valeur constante sera alors entrée dans l'entrée associée pour l'instruction CALCULATE.

Le fait d'entrer des constantes sous forme d'entrées vous permet de copier l'instruction CALCULATE à d'autres endroits dans votre programme utilisateur sans avoir à changer la fonction. Vous pouvez alors modifier les valeurs ou les variables des entrées pour l'instruction sans modifier la fonction.

Lorsque CALCULATE est exécuté et que toutes les opérations individuelles dans le calcul s'achèvent avec succès, ENO est égal à 1. Sinon, ENO est égal à 0.

Pour un exemple d'instruction CALCULATE, voir "Utilisez l'opération CALCULER pour une équation mathématique complexe (Page 46)".

6.3.5 Fonctionnement des temporisations

Le S7-1200 prend en charge les temporisations suivantes

- La temporisation TP génère une impulsion avec une durée préétablie.
- La temporisation TON règle la sortie (Q) sur ON après une temporisation préétablie.
- La temporisation TOF règle la sortie (Q) sur ON et remet la sortie sur OFF après une temporisation préétablie.
- La temporisation TONR règle la sortie (Q) sur ON après une temporisation préétablie. Le temps écoulé est accumulé sur plusieurs durées jusqu'à ce que l'entrée de réinitialisation (R) soit utilisée pour réinitialiser le temps écoulé.
- La bobine PT (temporisation préétablie) charge une nouvelle valeur temporelle préétablie dans la temporisation spécifiée.
- La bobine RT (temporisation de réinitialisation) réinitialise la temporisation spécifiée.

Pour CONT et LOG, ces opérations sont disponibles sous forme d'une boîte d'instruction ou d'une bobine de sortie.

Le nombre de temporisations que vous pouvez utiliser dans votre programme utilisateur est limité uniquement par la quantité de mémoire dans la CPU. Chaque temporisation utilise 16 octets de mémoire.

Chaque temporisation utilise une structure sauvegardée dans un bloc de données afin de conserver les données de la temporisation. Pour SCL, vous devez d'abord créer le DB pour l'opération de temporisation individuelle avant de pouvoir la référencer. Pour CONT et LOG, STEP 7 crée automatiquement le DB lorsque vous insérez l'opération.

Lorsque vous créez le DB, vous pouvez également utiliser un DB multi-instance. Comme les données de temporisation sont contenues dans un seul DB et qu'il n'est nécessaire d'avoir un DB séparé pour chaque temporisation, le temps de traitement des temporisations est réduit. Il n'y a pas d'interaction entre les structures des données de temporisation dans le DB multi-instance partagé.

Tableau 6- 9 TP (temporisation d'impulsion)

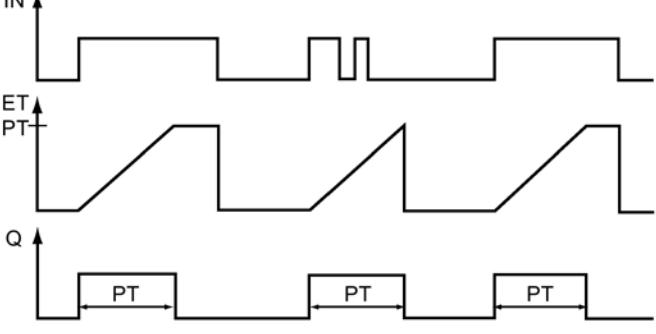
CONT/LOG	SCL	Chronogramme
 IEC_Timer_0 TP Time IN Q PT ET	<pre>"timer_db".TP(IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);</pre>	 <p>IN</p> <p>ET PT</p> <p>Q</p> <p>PT</p>

Tableau 6- 10TON (temporisation ON)

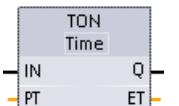
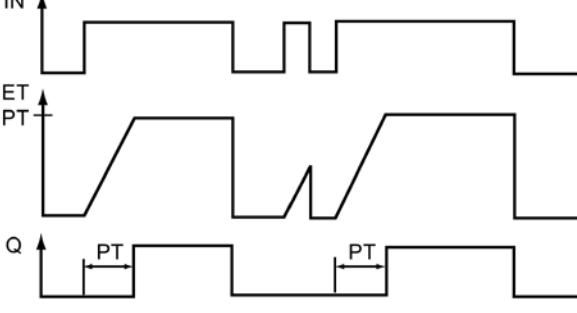
CONT/LOG	SCL	Chronogramme
 IEC_Timer_1 TON Time IN Q PT ET	<pre>"timer_db".TON(IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);</pre>	 <p>IN</p> <p>ET PT</p> <p>Q</p> <p>PT</p>

Tableau 6- 11TOF (temporisation OFF)

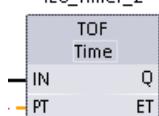
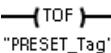
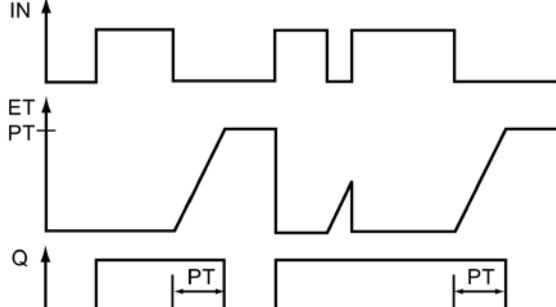
CONT/LOG	SCL	Chronogramme
 TOF_DB 	<pre>"timer_db".TOF(IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);</pre>	

Tableau 6- 12TONR (temporisation de rétention ON)

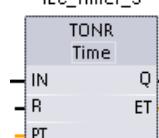
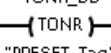
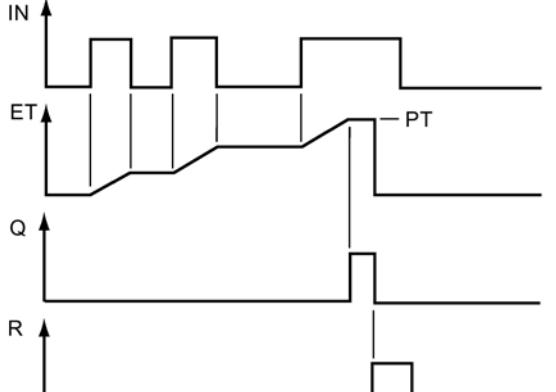
CONT/LOG	SCL	Chronogramme
 TONR_DB 	<pre>"timer_db".TONR(IN:=_bool_in_, R:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);</pre>	

Tableau 6- 13Opérations de bobine de temporisation préétablie -(PT)- et de temporisation réinitialisée -(RT)-

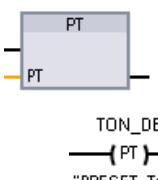
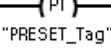
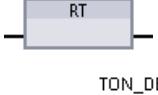
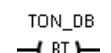
CONT/LOG	SCL	Description
 TON_DB 	<pre>PRESET_TIMER(PT:=_time_in_, TIMER:=_iec_timer_in_);</pre>	<p>Utilisez les opérations de bobine de temporisation préétablie -(PT)- et de temporisation réinitialisée -(RT)- avec les temporisations de boîte ou de bobine. Ces opérations de bobine peuvent être placées dans une position intermédiaire. L'état du flux de courant de sortie de la bobine est toujours le même que l'état d'entrée de la bobine.</p> <ul style="list-style-type: none"> • Lorsque la bobine -(PT)- est activée, l'élément de temps PRESET des données de DB IEC_Timer spécifiées est initialisé sur la durée "PRESET_TAG". • Lorsque la bobine -(RT)- est activée, l'élément de temps ELAPSED des données de DB IEC_Timer spécifiées est réinitialisé à 0.
 TON_DB 	<pre>RESET_TIMER(_iec_timer_in_);</pre>	

Tableau 6- 14 Types de données pour les paramètres

Paramètre	Type de données	Description
Boîte : ENTRÉE Bobine : Flux de courant	Bool	TP, TON et TONR : Boîte : 0 = désactiver la temporisation, 1 = activer la temporisation Bobine : pas de flux de courant = désactiver la temporisation, flux de courant = activer la temporisation TOF : Boîte : 0 = activer la temporisation, 1 = désactiver la temporisation Bobine : pas de flux de courant = activer la temporisation, flux de courant = désactiver la temporisation
L	Bool	Boîte TONR uniquement : 0 = pas de réinitialisation 1 = remettre le temps écoulé et le bit Q à 0
Boîte : PT Bobine : "PRESET_Tag"	Time	Boîte ou bobine de temporisation : Entrée valeur de temps prédéfinie
Boîte : Q Bobine : DBdata.Q	Bool	Boîte de temporisation : sortie de boîte Q ou bit Q dans les données du DB de temporisation Bobine de temporisation : vous pouvez accéder au bit Q uniquement dans les données du DB de temporisation
Boîte : ET Bobine : DBdata.ET	Time	Boîte de temporisation : sortie de boîte ET (temps écoulé) ou valeur de temps ET dans les données du DB de temporisation Bobine de temporisation : vous pouvez accéder à la valeur de temps ET uniquement dans les données du DB de temporisation.

Tableau 6- 15 Effet de modifications de valeurs des paramètres PT et IN

Temporisation	Modifications des paramètres de boîte PT et IN et des paramètres de bobine correspondants
TP	<ul style="list-style-type: none"> La modification de PT n'a aucun effet tant que la temporisation fonctionne. La modification de IN n'a aucun effet tant que la temporisation fonctionne.
TON	<ul style="list-style-type: none"> La modification de PT n'a aucun effet tant que la temporisation fonctionne. Le fait de passer IN sur FAUX, alors que la temporisation fonctionne, réinitialise et arrête la temporisation.
TOF	<ul style="list-style-type: none"> La modification de PT n'a aucun effet tant que la temporisation fonctionne. Le fait de passer IN sur VRAI, alors que la temporisation fonctionne, réinitialise et arrête la temporisation.
TONR	<ul style="list-style-type: none"> La modification de PT n'a aucun effet pendant que la temporisation fonctionne, mais a un effet lorsque la temporisation reprend. Le fait de passer IN sur FAUX, alors que la temporisation fonctionne, arrête mais ne réinitialise pas la temporisation. Le fait de repasser IN sur VRAI fera que la temporisation recommencera le chronométrage à la valeur de temps accumulé.

6.3 Les opérations puissantes facilitent la programmation

Les valeurs PT (temps préétabli) et ET (temps écoulé) sont stockées dans les données de DB IEC_TIMER spécifiées sous forme d'entiers doubles signés qui représentent des millisecondes de temps. Les données TIME utilisent l'identifiant T# et peuvent être entrées sous forme d'unités de temps simples (T#200ms ou 200) et sous forme d'unités de temps composées comme T#2s_200ms.

Tableau 6- 16 Dimension et portée du type de données de TIME

Type de données	Dimension	Portées de nombres valides ¹
TIME	32 bits, stockés comme données DInt	T#-24d_20h_31m_23s_648ms à T#24d_20h_31m_23s_647ms Stockés comme -2 147 483 648 ms à +2 147 483 647 ms

¹ La portée négative du type de données TIME montrée ci-dessus ne peut pas être utilisée avec les opérations de temporisation. Les valeurs de PT (temps préétabli) négatives sont réglées à zéro lorsque l'opération de temporisation est exécutée. ET (temps écoulé) est toujours une valeur positive.

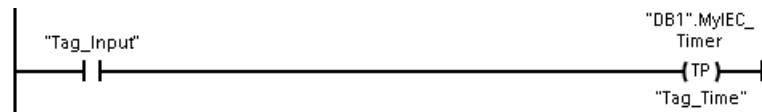
Programmation des temporisations

Vous devez tenir compte des conséquences suivantes du fonctionnement des temporisations lors de la planification et de la création de votre programme utilisateur :

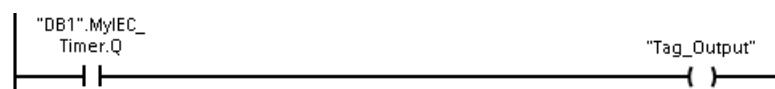
- Plusieurs actualisations d'une temporisation peuvent avoir lieu dans le même cycle. La temporisation est actualisée à chaque fois que l'instruction de temporisation (TP, TON, TOF, TONR) est exécutée et à chaque fois que l'élément ELAPSED ou Q de la structure de temporisation est utilisé comme paramètre d'une autre instruction exécutée. Cela constitue un avantage si vous voulez disposer des dernières données de temps (une lecture directe de la temporisation essentiellement). Toutefois, si vous désirez avoir des valeurs cohérentes durant tout un cycle de programme, vous devez placer votre instruction de temporisation avant toutes les autres instructions ayant besoin de ces valeurs et utiliser des variables connectées aux sorties Q et ET de l'instruction de temporisation plutôt que les éléments ELAPSED et Q de la structure de DB de temporisation.
- Il peut y avoir des cycles pendant lesquels une temporisation n'est pas actualisée. Il est possible de démarrer votre temporisation dans une fonction et de ne plus appeler cette fonction pendant un ou plusieurs cycles. Si aucune autre instruction référençant les éléments ELAPSED ou Q de la structure de temporisation n'est exécutée, la temporisation n'est pas actualisée. Il n'y aura pas d'actualisation tant que l'instruction de temporisation n'est pas réexécutée ou qu'une autre instruction utilisant l'élément ELAPSED ou Q de la structure de temporisation comme paramètre n'est pas exécutée.
- Bien que cela ne soit pas typique, vous pouvez affecter la même structure de temporisation de DB à plusieurs instructions de temporisation. Mais en général, pour éviter des interactions imprévisibles, vous n'utiliserez qu'une instruction de temporisation (TP, TON, TOF, TONR) par structure de temporisation de DB.

Les temporisations à auto-réinitialisation sont utiles pour déclencher des actions qui doivent se produire périodiquement. Typiquement, vous créez des temporisations à auto-réinitialisation en plaçant un contact à ouverture qui référence le bit de temporisation devant l'instruction de temporisation. Ce réseau de temporisation est typiquement situé au-dessus d'un ou de plusieurs réseaux dépendants qui utilisent le bit de temporisation pour déclencher des actions. Lorsque la temporisation expire (le temps écoulé atteint la valeur prédéfinie), le bit de temporisation est activé pour un cycle, permettant ainsi à la logique de réseau dépendante pilotée par le bit de temporisation de s'exécuter. Lors de l'exécution suivante du réseau de temporisation, le contact à ouverture est désactivé, ce qui réinitialise la temporisation et efface le bit de temporisation. Au cycle suivant, le contact à ouverture est activé ce qui redémarre la temporisation. Lors de la création d'une telle temporisation à auto-réinitialisation, n'utilisez pas l'élément "Q" de la structure de DB de temporisation comme paramètre pour le contact à ouverture devant l'instruction de temporisation, mais utilisez la variable connectée à la sortie "Q" de l'instruction de temporisation prévue à cet effet. En effet, il ne faut pas accéder à l'élément Q de la structure de temporisation de DB, car cela provoque une actualisation de la temporisation et que, si la temporisation est actualisée en raison du contact à ouverture, le contact réinitialisera l'instruction de temporisation immédiatement. La sortie Q de l'instruction de temporisation ne sera pas activée pour un cycle et les réseaux dépendants ne s'exécuteront pas.

Les bobines de temporisation -(TP)-, -(TON)-, -(TOF)- et -(TONR)- doivent correspondre à la dernière opération d'un réseau. Comme indiqué dans l'exemple de temporisation, une opération de contact dans un réseau subséquent évalue le bit Q dans les données DB IEC_Timer d'une bobine de temporisation. De même, vous devez aborder l'élément ELAPSED dans les données DB IEC_timer si vous souhaitez utiliser la valeur de temps écoulé dans votre programme.



La temporisation d'impulsion commence à un passage de 0 à 1 de la valeur de bit de Tag_Input. La temporisation fonctionne pendant le temps spécifié par la valeur de temps Tag_Time.



Tant que la temporisation fonctionne, l'état du DB1.MyIEC_Timer.Q=1 et la valeur Tag_Output =1. Lorsque la valeur Tag_Time est écoulée, DB1.MyIEC_Timer.Q=0 et la valeur Tag_Output =0.

6.3.6

Fonctionnement des compteurs

Vous utilisez les compteurs pour compter des événements de programme internes et des événements de processus externes.

- Le compteur "croissant" (CTU) avance de 1 lorsque la valeur du paramètre d'entrée CU passe de 0 à 1.
- Le compteur "décroissant" (CTD) recule de 1 lorsque la valeur du paramètre d'entrée CD passe de 0 à 1.
- Le compteur "croissant et décroissant" (CTUD) compte en avançant ou en reculant de 1 sur le passage de 0 à 1 des entrées croissantes (CU) ou décroissantes (CD).

S7-1200 fournit également des compteurs rapides (Page 135) (HSC) pour compter les événements qui surviennent plus rapidement que le temps d'exécution de l'OB.

Les opérations CU, CD et CTUD utilisent les compteurs logiciels dont la fréquence de comptage maximal est limité par le temps d'exécution de l'OB dans lequel elles sont placées.

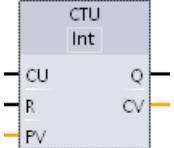
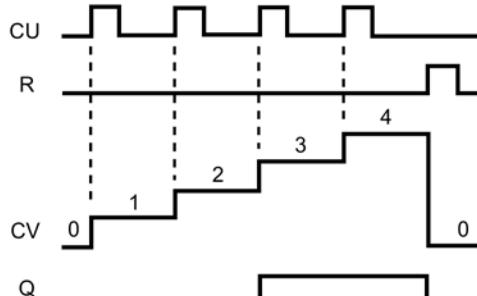
Remarque

Si les événements à compter surviennent dans le temps d'exécution de l'OB, utilisez les opérations de compteur CTU, CTD ou CTUD. Si les événements surviennent plus rapidement que le temps d'exécution de l'OB, utilisez le HSC.

Chaque compteur utilise une structure sauvegardée dans un bloc de données afin de conserver les données du compteur. Pour SCL, vous devez d'abord créer le DB pour l'opération de compteur individuelle avant de pouvoir le référencer. Pour CONT et LOG, STEP 7 crée automatiquement le DB lorsque vous insérez l'opération.

Le nombre de compteurs que vous pouvez utiliser dans votre programme utilisateur est limité uniquement par la mémoire disponible dans la CPU. Les compteurs individuels utilisent 3 octets (pour SInt ou USInt), 6 octets (pour Int ou UInt) ou 12 octets (pour DInt ou UDInt).

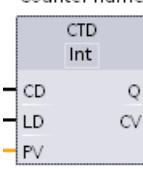
Tableau 6- 17 Compteur CTU (croissant)

CONT/LOG	SCL	Opération
<p>"Counter name"</p> 	<pre>"ctu_db".CTU(CU:=_bool_in, R:=_bool_in, PV:=_in_, Q=>_bool_out, CV=>_out_);</pre>	

Le chronogramme illustre le fonctionnement d'un compteur CTU avec une valeur de comptage entière non signée (où PV = 3).

- Si la valeur de paramètre CV (valeur de comptage en cours) est supérieure ou égale à la valeur de paramètre PV (valeur de comptage préétablie), alors le paramètre de sortie de compteur Q = 1.
- Si la valeur du paramètre réinitialisé R passe de 0 à 1, alors CV est réinitialisé à 0.

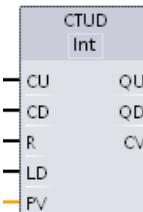
Tableau 6- 18 Compteur CTD (décroissant)

CONT/LOG	SCL	Fonctionnement
<p>"Counter name"</p> 	<pre>"ctd_db".CTD(CD := _bool_in, LD := _bool_in, PV := _in_, Q=>_bool_out, CV=>_out_);</pre>	

Le chronogramme illustre le fonctionnement d'un compteur CTD avec une valeur de comptage entière non signée (où PV = 3).

- Si la valeur de paramètre CV (valeur de comptage en cours) est égale ou inférieure à 0, le paramètre de sortie de comptage Q = 1.
- Si la valeur du paramètre LD passe de 0 à 1, la valeur au paramètre PV (valeur préétablie) est chargée sur le compteur comme nouvelle CV.

Tableau 6- 19 Compteur CTUD (croissant et décroissant)

CONT/LOG	SCL	Fonctionnement
<p>"Counter name"</p> 	<pre>"ctud_db".CTUD(CU := _bool_in, CD := _bool_in, R := _bool_in, LD := _bool_in, PV := _in_, QU=>_bool_out, QD=>_bool_out, CV=>_out_);</pre>	

Le chronogramme illustre le fonctionnement d'un compteur CTUD avec une valeur de comptage entière non signée (où PV = 4).

- Si la valeur de paramètre CV (valeur de comptage en cours) est supérieure ou égale à la valeur de paramètre PV (valeur préétablie), alors le paramètre de sortie de compteur QU = 1.
- Si la valeur de paramètre CV est inférieure ou égale à zéro, alors le paramètre de sortie de compteur QD = 1.
- Si la valeur du paramètre LD passe de 0 à 1, la valeur au paramètre PV est chargée sur le compteur comme nouvelle CV.
- Si la valeur du paramètre réinitialisé R passe de 0 à 1, alors CV est réinitialisé à 0.

6.3.7 Modulation de largeur d'impulsion (PWM)

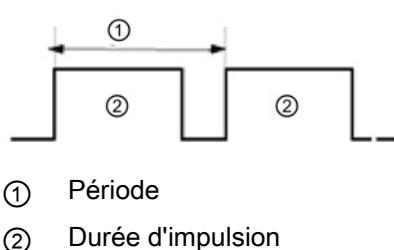
L'instruction CTRL_PWM est disponible dans le groupe Impulsion des Instructions avancées.

Tableau 6- 20 Instruction CTRL_PWM

CONT/LOG	SCL	Description
 <p>"CTRL_PWM_DB"</p> <p>CTRL_PWM</p> <ul style="list-style-type: none"> EN → ENO PWM → BUSY ENABLE → STATUS 	<pre>"ctrl_pwm_db"(PWM:=W#16#0, ENABLE:=False, BUSY=>_bool_out_, STATUS=>_word_out_);</pre>	<p>L'instruction CTRL_PWM fournit une sortie de période fixe avec un rapport de cycle variable. La sortie PWM s'exécute en continu une fois qu'elle a été lancée à la fréquence indiquée (période). La largeur d'impulsion est variée comme requis pour attribuer la commande souhaitée.</p>

Lorsque vous insérez l'instruction CTRL_PWM dans votre bloc de code, vous créez le DB de l'instruction dans la boîte de dialogue "Options d'appel". L'instruction CTRL_PWM stocke les informations de paramètres dans le DB et commande les paramètres du bloc de données.

La largeur d'impulsion sera réglée à la valeur initiale configurée dans la configuration de l'appareil lorsque la CPU entre pour la première fois en mode MARCHE. Vous saisissez les valeurs à l'adresse de sortie (Q) de la longueur de mots qui était spécifiée dans la configuration de l'appareil ("adresses de sortie" / "adresse de démarrage") comme requis pour modifier la largeur d'impulsion. Utilisez une instruction (par exemple Move, Convert, Math, ou PID) pour inscrire la largeur d'impulsion spécifiée à la sortie de longueur de mots appropriée (Q). Vous devez utiliser la plage valide pour la valeur de sortie (pourcentage, millièmes, dix millièmes ou format analogique S7).



Le rapport cycle peut être exprimé, par exemple, sous forme de pourcentage de la période ou sous forme de quantité relative (telle que 0 pour 1000 ou 0 pour 10000). La largeur d'impulsion peut varier de 0 (pas d'impulsion, toujours désactivée) à la pleine échelle (pas d'impulsion, toujours activée).

La sortie PWM peut varier de 0 à la pleine échelle en fournissant une sortie numérique qui est, à de nombreux égards, la même qu'une sortie analogique. Par exemple, la sortie PWM peut être utilisée pour contrôler la vitesse d'un moteur de l'arrêt à la pleine vitesse ou pour contrôler la position d'une valve de fermée à grande ouverte.

6.4

Enregistrements de données faciles à créer

Votre programme de commande peut utiliser les instructions Data log pour stocker des valeurs de données d'exécution dans des fichiers journaux permanents. Les fichiers journaux sont stockés en mémoire flash (CPU ou carte mémoire). Les données des journaux sont stockées en format CSV standard (valeurs séparées par une virgule). Les enregistrements sont organisés sous forme de fichier journal circulaire d'une dimension pré-déterminée.

Vous utiliserez les instructions Data log dans votre programme pour créer, ouvrir, écrire un enregistrement et fermer les fichiers journaux. Vous décidez des valeurs du programme qui seront consignées en créant une mémoire tampon qui définit un enregistrement de journal unique. Votre mémoire tampon de données sert de stockage temporaire pour un nouvel enregistrement de journal. Les nouvelles valeurs en cours doivent être copiées par programme dans la mémoire tampon pendant l'exécution. Une fois toutes les valeurs de données en cours actualisées, vous pouvez exécuter l'instruction DataLogWrite pour transférer les données de la mémoire tampon dans un enregistrement de journal.

Vous pouvez ouvrir, éditer, enregistrer, renommer et supprimer des fichiers journaux de la page Navigateur de Fichiers du Serveur Web. Vous devez avoir les droits pour visualiser le navigateur de fichiers et vous devez avoir modifié les droits pour éditer, supprimer ou renommer des fichiers journaux.

6.4 Enregistrements de données faciles à créer

Utilisez les instructions DataLog pour stocker les données de process d'exécution par programme dans la mémoire flash de la CPU. Les enregistrements de données sont organisés sous forme d'un fichier d'enregistrement circulaire d'une dimension prédéterminée. Les nouveaux enregistrements sont ajoutés au fichier d'enregistrement des données. Une fois que le fichier d'enregistrements a stocké le nombre maximal d'enregistrements, l'enregistrement écrit suivant remplace le plus ancien. Pour empêcher que des enregistrements en remplacement d'autres, utiliser l'instruction DataLogNewFile. Les nouveaux enregistrements de données sont stockés dans le nouveau fichier d'enregistrement des données, tandis que l'ancien fichier d'enregistrement reste dans la CPU.

Tableau 6- 21 Instruction DataLogWrite

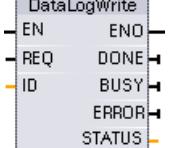
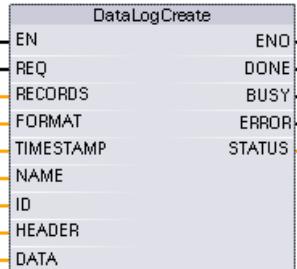
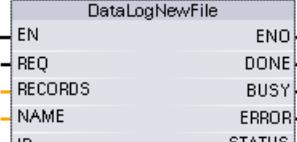
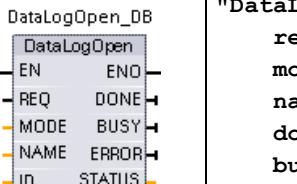
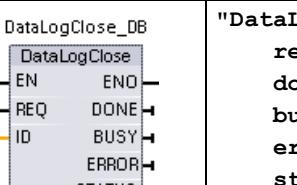
CONT/LOG	SCL	Description
	<pre>"DataLogWrite_DB" (req:=FALSE, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, ID:=_dword_inout_);</pre>	<p>DataLogWrite écrit un enregistrement de données dans l'enregistrement spécifié. L'enregistrement de données cible existant doit être ouvert.</p> <p>Vous devez charger la mémoire tampon par programme avec les valeurs de données en cours et exécuter ensuite l'instruction DataLogWrite pour déplacer les nouvelles données de la mémoire tampon au fichier d'enregistrement.</p> <p>En cas de panne de courant lors d'une instruction DataLogWrite inachevée, l'enregistrement des données transféré vers l'enregistrement de données pourrait être perdu.</p>

Tableau 6- 22 Instructions DataLogCreate et DataLogNewFile

CONT/LOG	SCL	Description
	<pre>"DataLogCreate_DB"(req:=FALSE, records:=1, format:=1, timestamp:=1, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, name:=_variant_in_, ID:=_dword_inout_, header:=_variant_inout_, data:= variant inout_);</pre>	DataLogCreate ¹ crée et initialise un fichier d'enregistrement de données stocké dans le répertoire \DataLogs de la CPU. Le fichier d'enregistrement des données est créé dans une taille fixe pré-déterminée.
	<pre>"DataLogNewFile_DB"(req:=FALSE, records:=1, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, name:=_variant_in_, ID:=_dword_inout_);</pre>	DataLogNewFile ¹ permet à votre programme de créer un nouveau fichier journal sur la base d'un fichier journal existant. Un nouveau fichier d'enregistrement des données sera créé et implicitement ouvert en se basant sur le NAME spécifié. L'enregistrement en-tête sera dupliqué à partir du fichier de données original avec les propriétés du fichier original. Le fichier d'enregistrement des données original sera implicitement fermé.

- ¹ L'exécution d'une instruction DataLogNewFile s'étend sur de nombreux cycles du programme. La durée réelle nécessaire à la création d'un fichier journal dépend de la structure d'un enregistrement et du nombre d'enregistrements. Avant que le nouveau fichier journal puisse être utilisé pour d'autres opérations d'enregistrement de données, la logique de votre programme doit surveiller la transition du bit DONE à TRUE.

Tableau 6- 23 Instructions DataLogOpen et DataLogClose

CONT/LOG	SCL	Description
	<pre>"DataLogOpen_DB"(req:=FALSE, mode:=0, name:=_variant_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, ID:=_dword_inout_);</pre>	L'instruction DataLogOpen ouvre un fichier journal pré-existent. Il faut qu'un journal soit ouvert pour que vous puissiez y écrire de nouveaux enregistrements. Les journaux peuvent être ouverts et fermés de façon individuelle. Huit journaux peuvent être ouverts en même temps.
	<pre>"DataLogClose_DB"(req:=FALSE, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, ID:=_dword_inout_);</pre>	L'instruction DataLogClose ferme un fichier journal ouvert. Des opérations DataLogWrite sur un journal fermé provoquent une erreur. Aucune opération d'écriture n'est autorisée sur ce journal tant qu'une autre opération DataLogOpen n'a pas été exécutée. Un passage à l'état ARRÊT ferme tous les fichiers journaux ouverts.

6.5 Facilité de surveillance et d'essai de votre programme utilisateur

6.5.1 Tables de visualisation et tables de forçage permanent

Vous pouvez utiliser des "tables de visualisation" pour visualiser et modifier les valeurs d'un programme utilisateur exécuté par la CPU en ligne. Vous pouvez créer et sauvegarder différentes tables de visualisation dans votre projet afin de prendre en charge des environnements de test variés. Vous pouvez ainsi reproduire des tests pendant la mise en service ou à des fins de dépannage et de maintenance.

Avec une table de visualisation, vous pouvez surveiller la CPU et interagir avec elle alors qu'elle exécute le programme utilisateur. Vous pouvez afficher ou changer les valeurs non seulement pour les variables des blocs de code et de données, mais également pour les zones de mémoire de la CPU, à savoir les entrées et sorties (I et Q), la périphérie d'entrée (I:P), les mémentos (M) et les blocs de données (DB).

Avec la table de visualisation, vous pouvez débloquer les sorties physiques (Q:P) d'une CPU à l'état ARRET. Vous pouvez, par exemple, affecter des valeurs spécifiques aux sorties lorsque vous testez le câblage de la CPU.

STEP 7 fournit également une table de forçage permanent pour le forçage permanent d'une variable à une valeur spécifique. Pour plus d'informations sur le forçage, reportez-vous au paragraphe traitant du forçage de valeurs dans la CPU (Page 359) au chapitre "Outils en ligne et de diagnostic".

Remarque

Les valeurs de forçage permanent sont stockées dans la CPU et non dans la table de visualisation.

Vous ne pouvez pas forcer une entrée (adresse I), mais vous pouvez forcer une entrée de périphérie. Pour forcer une entrée de périphérie, ajoutez ":P" à l'adresse (par exemple, "On:P").

STEP 7 permet également de tracer et d'enregistrer des variables du programme en fonction de conditions de déclenchement (Page 374).

6.5.2 Affichage de l'usage des références croisées

La fenêtre d'inspection affiche des informations sous forme de références croisées indiquant la manière dont un objet sélectionné est utilisé dans l'ensemble du projet, comme p. ex. le programme utilisateur, la CPU ou tout appareil IHM. La table des "Références croisées" affiche les instances où un objet sélectionné et utilisé, de même que les autres objets qui l'utilisent. La fenêtre d'inspection contient également les blocs qui sont exclusivement accessibles en ligne dans les références croisées. Pour afficher les références croisées, choisissez la commande "Afficher références croisées" (dans la vue du projet, les références croisées font partie du menu "Outils").

Remarque

Il n'est pas nécessaire de fermer l'éditeur pour voir les informations sur les références croisées.

Vous avez la possibilité de trier les entrées des références croisées. La liste des références croisées fournit une vue d'ensemble de l'utilisation des adresses mémoire et des variables dans le programme utilisateur.

- Lorsque vous créez et modifiez un programme, vous obtenez une vue d'ensemble des opérandes, variables et appels de blocs utilisés.
- Vous pouvez, à partir des références croisées, sauter directement à l'occurrence d'utilisation des opérandes et variables.
- Durant un test du programme ou lors d'un dépannage, vous voyez quelle adresse de mémoire est traitée par quelle commande et dans quel bloc, quelle variable est utilisée dans quelle vue et quel bloc est appelé par quel autre bloc.

Tableau 6- 24 Éléments de la référence croisée

Colonne	Description
Objet	Nom de l'objet utilisant les objets de niveau inférieur ou qui est utilisé par les objets de niveau inférieur
Nombre	Nombre d'utilisations
Endroit d'utilisation	Toute adresse possible, p. ex. adresse de réseau.
Propriétés	Propriétés spécifiques des objets référencés, p. ex. noms de variables dans des déclarations multiinstances
En tant que	Affiche des informations supplémentaires sur l'objet, p. ex. si un DB d'instance est utilisé en tant que modèle ou instance multiple.
Accès	Type d'accès à l'opérande, à savoir en lecture (R) et/ou en écriture (W).
Adresse	Adresse de l'opérande
Type	Information sur le type et le langage utilisés pour créer l'objet.
Chemin	Chemin de l'objet dans l'arborescence

Selon les produits installés, le tableau des références croisées affiche des colonnes supplémentaires ou différentes.

6.5.3

Structure d'appel permettant de constater la hiérarchie d'appel

La structure d'appel décrit la hiérarchie d'appel des blocs dans votre programme utilisateur. Elle fournit une vue d'ensemble des blocs utilisés, des appels d'autres blocs, des relations entre blocs, des données requises pour chaque bloc et de l'état des blocs. Vous pouvez ouvrir l'éditeur de programmes et éditer les blocs de la structure d'appel.

L'affichage de la structure d'appel fournit une liste des blocs utilisés dans le programme utilisateur. STEP 7 met en évidence le premier niveau de la structure d'appel et affiche tout bloc n'étant pas appelé par un autre bloc dans le programme. Le premier niveau de la structure d'appel affiche les OB ainsi que tous les FC, FB et DB qui ne sont pas appelés par un OB. Si un bloc de code appelle un autre bloc, le bloc appelé est représenté en retrait sous le bloc appelant. La structure d'appel affiche uniquement les blocs qui sont appelés par un blocs de code.

Vous avez la possibilité d'afficher exclusivement les blocs entraînant des conflits dans la structure d'appel. Les conditions suivantes entraînent des conflits :

- Blocs exécutant des appels avec des horodatages plus anciens ou plus récents
- Blocs appelant un bloc avec une interface modifiée
- Blocs utilisant une variable avec une adresse et/ou un type de données modifiés
- Blocs n'étant pas appelés directement ou indirectement par un OB.
- Blocs appelant un bloc inexistant ou manquant

Vous pouvez regrouper plusieurs appels de bloc et blocs de données. Une liste déroulante indique les liens aux différentes adresses d'appel.

Vous pouvez également effectuer un contrôle de cohérence pour afficher les conflits d'horodatage. Un changement d'horodatage d'un bloc durant ou après la compilation du programme risque d'entraîner des conflits d'horodatage, qui à leur tour provoquent des incohérences parmi les blocs appelants et les blocs appelés.

- Une recompilation des blocs de code permet de corriger la plupart des conflits d'horodatage et d'interface.
- Si la compilation ne permet pas de remédier aux incohérences, servez-vous du lien dans la colonne "Détails" pour aller à la source du problème dans l'éditeur de programmes. Vous pouvez alors corriger manuellement toutes les incohérences.
- Tous les blocs marqués en rouge doivent être recompilés.

6.5.4 Opérations de diagnostic pour contrôler le matériel

6.5.4.1 Lecture de l'état des DEL sur la CPU

L'instruction LED permet à votre programme utilisateur de déterminer l'état des DEL sur la CPU. Vous pouvez utiliser ces informations pour programmer une variable pour votre appareil IHM.

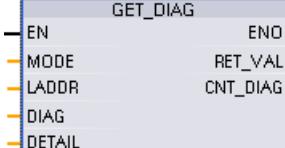
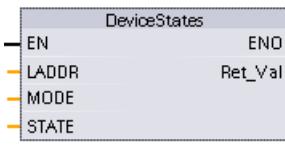
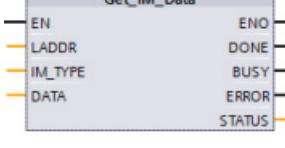
Tableau 6- 25 Opération LED

CONT/LOG	SCL	Description
	<pre>ret_val := LED(laddr:=_word_in_, LED:=_uint_in_);</pre>	<p>RET_VAL rapporte les états de DEL suivants pour la CPU</p> <ul style="list-style-type: none"> • MARCHE/ARRÊT : vert ou jaune • Erreur : rouge • MAINT (maintenance) : jaune • Liaison : vert • Tx/Rx (transmet/reçoit) : jaune

6.5.4.2 Instructions pour la lecture de l'état de diagnostic des appareils

STEP 7 inclut également des opérations pour la lecture des informations sur l'état fournies par les périphériques sur votre réseau.

Tableau 6- 26 Opérations de diagnostic

CONT/LOG	SCL	Description
	<pre>ret_val := GET_DIAG(mode:=_uint_in_, laddr:=_word_in_, cnt_diag=>_uint_out_, diag:=_variant_inout_, de- tail:= variant inout);</pre>	L'opération GET_DIAG lit les informations de diagnostic à partir du périphérique spécifié.
	<pre>ret_val := DeviceStates(laddr:=_word_in_, mode:=_uint_in_, state:=_variant_inout_);</pre>	L'instruction DeviceStates lit l'état des appareils PROFINET ou PROFIBUS.
	<pre>ret_val := ModuleStates(laddr:=_word_in_, mode:=_uint_in_, state:=_variant_inout_);</pre>	L'instruction ModuleStates lit l'état des modules PROFINET ou PROFIBUS.
	<pre>"GET_IM_DATA_DB" (LADDR:=16#0, IM_TYPE:=0, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, DATA:= variant inout);</pre>	Utilisez l'instruction Get_IM_Data pour vérifier les données d'identification et maintenance (I&M) pour le module ou sous-module spécifié.

6.6 Compteur rapide (HSC)

Utilisez des compteurs rapides (HSC) pour compter les événements qui surviennent plus rapidement que le taux d'exécution d'OB. Les instructions de comptage se trouvent dans la partie Technologie de l'arborescence d'instructions. L'instruction CTRL_HSC contrôle le fonctionnement du HSC.

Remarque

Si les événements à compter surviennent dans le taux d'exécution de l'OB, utilisez les instructions de compteur CTU, CTD, ou CTUD. Si les événements surviennent plus rapidement que le taux d'exécution, utilisez le HSC.

Vous configurez les paramètres pour chaque HSC dans la configuration de l'appareil pour la CPU : mode comptage, connexions E/S, attribution d'alarme et fonctionnement en tant que compteur rapide ou en tant qu'appareil de mesure de fréquence ou période d'impulsion.

Tableau 6- 27 Instruction CTRL_HSC

CONT/LOG	SCL	Description
 <pre> "counter_name" (HSC:=W#16#0, DIR:=FALSE, CV:=FALSE, RV:=FALSE, Period:=FALSE, New_DIR:=0, New_CV:=L#0, New_RV:=L#0, New_Period:=0, Busy=>_bool_out_, Status=>_word_out_); </pre>	<pre> "counter_name" (HSC:=W#16#0, DIR:=FALSE, CV:=FALSE, RV:=FALSE, Period:=FALSE, New_DIR:=0, New_CV:=L#0, New_RV:=L#0, New_Period:=0, Busy=>_bool_out_, Status=>_word_out_); </pre>	<p>Chaque instruction CTRL_HSC utilise une structure stockée dans un bloc de données pour conserver les données de compteur.</p> <p>Pour SCL, vous devez d'abord créer le DB pour l'instruction de compteur individuelle avant de pouvoir la référencer. Pour CONT et LOG, STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.</p>

L'instruction CTRL_HSC est en règle générale placée dans un OB d'alarme matérielle qui est exécuté lorsque l'événement d'alarme matérielle du compteur est déclenchée. Par exemple, si un événement CV=RV déclenche l'alarme du compteur, un bloc de code d'OB d'alarme matérielle exécute l'instruction CTRL_HSC et peut modifier la valeur de référence en chargeant une valeur NEW_RV.

Remarque

La valeur de comptage en cours n'est pas disponible dans les paramètres CTRL_HSC. L'adresse de l'image de process qui stocke la valeur de comptage en cours est attribuée lors de la configuration matérielle du compteur rapide. Vous pouvez utiliser la logique de programme pour lire directement la valeur de compte. La valeur renvoyée à votre programme sera un comptage correct pour l'instant où le compteur a été lu. Le compteur continuera à compter les événements rapides. Ainsi, la valeur de comptage en cours pourrait changer avant que votre programme n'ait terminé un processus basé sur une ancienne valeur de comptage.

Certains des paramètres pour le HSC peuvent être modifiés par votre programme utilisateur pour fournir un contrôle de programme du processus de comptage :

- Réglez la direction de comptage sur une valeur NEW_DIR
- Réglez la valeur de comptage en cours sur une valeur NEW_CV
- Réglez la valeur de référence sur une valeur NEW_RV
- Réglez la valeur de période (pour le mode mesure de fréquence) sur une valeur NEW_PERIOD

Si les valeurs drapeaux booléennes suivantes sont réglées à 1 lorsque l'instruction CTRL_HSC est exécutée, la valeur NEW_xxx correspondante est chargée sur le compteur. De multiples requêtes (plusieurs drapeaux sont établis au même moment) sont traitées en une seule exécution de l'instruction CTRL_HSC. Le réglage des valeurs de signalisation booléennes sur 0 ne produira aucune modification.

- Réglage de DIR = 1 charge une valeur NEW_DIR.
- Réglage de CV = 1 charge une valeur NEW_CV.
- Réglage de RV = 1 charge une valeur NEW_RV.
- Réglage de PERIOD = 1 charge une valeur NEW_PERIOD.

Instruction CTRL_HSC_EXT (instruction (avancée) Commander les compteurs rapides)

STEP 7 et la CPU S7-1200 prennent également en charge une instruction avancée de compteur rapide, CTRL_HSC_EXT. Cette instruction permet au programme de mesurer précisément la période des impulsions d'entrée d'un HSC désigné. Pour plus de détails, référez-vous au *Manuel système de l'automate programmable S7-1200*.

6.6.1

Fonctionnement du compteur rapide

Les compteurs rapides (HSC) peuvent compter des événements se produisant plus rapidement que la vitesse d'exécution de l'OB cyclique. Si les événements à compter surviennent en deçà de la vitesse d'exécution de l'OB, vous pouvez utiliser des opérations de comptage standard CTU, CTD ou CTUD. Si les événements surviennent plus rapidement que la vitesse d'exécution de l'OB, utilisez le compteur HSC plus rapide. L'instruction CTRL_HSC autorise la modification de certains des paramètres HSC par le programme.

Vous pouvez, par exemple, utiliser un compteur rapide comme entrée pour un codeur angulaire incrémental. Le codeur angulaire fournit un nombre défini de valeurs de comptage par tour et une impulsion de réinitialisation par tour. Les horloges et l'impulsion de réinitialisation du codeur angulaire constituent les entrées du compteur rapide.

La première de plusieurs valeurs prédéfinies est chargée dans le compteur rapide et les sorties sont activées pour la durée où la valeur de comptage en cours est inférieure à la valeur prédéfinie en vigueur. Le compteur rapide fournit une alarme lorsque la valeur en cours est égale à la valeur prédéfinie, lorsqu'une réinitialisation se produit et lorsque le sens de comptage change.

Une nouvelle valeur prédéfinie est chargée et l'état suivant des sorties est activé à chaque fois que l'événement d'alarme "Valeur en cours égale à valeur prédéfinie" se produit. Lorsque l'événement d'alarme "Réinitialisation" survient, la première valeur prédéfinie ainsi que le premier état des sorties sont activés et le cycle se répète.

Comme les alarmes se produisent à une fréquence bien inférieure à la vitesse de comptage du compteur rapide, il est possible de réaliser une commande précise des opérations rapides qui a un impact relativement mineur sur le cycle de la CPU. La méthode d'association d'alarmes à des programmes d'alarme permet d'effectuer chaque chargement d'une nouvelle valeur prédéfinie dans un programme d'alarme distinct, ce qui facilite la commande. Mais vous pouvez aussi traiter tous les événements d'alarme dans un programme d'alarme unique.

Sélection de canal d'entrée HSC

Utilisez le tableau suivant et assurez-vous que les canaux d'entrée de la CPU et du SB que vous connectez peuvent prendre en charge les vitesses d'impulsions maximales dans vos messages du processus.

Remarque

Les canaux d'entrée de la CPU et du SB (firmware V4 ou ultérieur) ont des valeurs de temps de filtre d'entrée configurables

Les versions plus anciennes du firmware avaient des canaux d'entrée HSC fixes et des valeurs de temps de filtre fixes qui ne pouvaient pas être modifiés.

Les versions V4 ou supérieures vous permettent d'affecter des valeurs de temps de canal d'entrée et de filtre. Le réglage de filtre d'entrée par défaut de 6,4 ms peut être trop lent pour vos messages de processus. Vous devez optimiser les valeurs de temps de filtre d'entrée TOR pour les entrées HSC pour votre application HSC.

Tableau 6- 28 entrée CPU : fréquence maximale

CPU	canal d'entrée CPU	mode phase 1 ou 2	mode phase quadrature A/B
1211C	la.0 à la.5	100 kHz	80 kHz
1212C	la.0 à la.5	100 kHz	80 kHz
	la.6, la.7	30 kHz	20 kHz
1214C et 1215C	la.0 à la.5	100kHz	80kHz
	la.6 à lb.5	30 kHz	20 kHz
1217C	la.0 à la.5	100 kHz	80 kHz
	la.6 à lb.1	30 kHz	20 kHz
	lb.2 à lb.5 (.2+, .2- à .5+, .5-)	1 MHz	1 MHz

Tableau 6- 29 entrée de signal board SB : fréquence maximale (carte facultative)

Signal board SB	canal d'entrée de SB	mode phase 1 ou 2	mode phase quadrature A/B
SB 1221, 200 kHz	le.0 à le.3	200kHz	160 kHz
SB 1223, 200 kHz	le.0, le.1	200kHz	160 kHz
SB 1223	le.0, le.1	30 kHz	20 kHz

Sélection des fonctions du HSC

Tous les compteurs rapides fonctionnent de la même manière dans un mode donné. Le mode de compteur, la commande de direction et la direction initiale sont affectés dans la configuration d'appareil de CPU pour les propriétés de fonction HSC.

Il existe quatre types fondamentaux de compteurs rapides :

- Compteur monophase avec gestion interne du sens de comptage
- Compteur monophase avec gestion externe du sens de comptage
- Compteur biphasé avec deux entrées d'horloge
- Compteur en quadrature de phase A/B

Vous pouvez utiliser chaque type de compteur rapide avec ou sans entrée de réinitialisation. Si vous activez l'entrée de réinitialisation (avec certaines restrictions, voir le tableau suivant), la valeur en cours est effacée et le reste jusqu'à ce que vous désactiviez l'entrée de réinitialisation.

- Fonction fréquence : Certains modes HSC permettent de configurer (type de comptage) le compteur rapide afin qu'il renvoie la fréquence et non une valeur de comptage d'impulsions en cours. Trois périodes de mesure de fréquence différentes sont disponibles : 0,01, 0,1 ou 1,0 seconde.

La période de mesure de fréquence détermine à quel rythme le compteur rapide calcule et renvoie une nouvelle valeur de fréquence. La fréquence renvoyée est une valeur moyenne déterminée par le nombre total de comptages pendant la dernière période de mesure. Si la fréquence change rapidement, la valeur renvoyée est une valeur intermédiaire entre la fréquence la plus élevée et la fréquence la plus faible apparaissant pendant la période de mesure. La fréquence est toujours indiquée en hertz (impulsions par seconde) quel que soit le paramétrage de la période de mesure de fréquence.

- Modes et entrées du compteur : Le tableau suivant montre les entrées utilisées pour les fonctions d'horloge, de gestion du sens de comptage et de réinitialisation associées aux compteurs rapides.
- Fonction de mesure de la période : Une mesure de la période est fournie avec l'intervalle de mesure configuré (10ms, 100ms, ou 1000ms). HSC_Period SDT renvoie des mesures de la période et fournit les mesures de la période sous forme de deux valeurs : ElapsedTime et EdgeCount. Les entrées ID1000 à ID1020 de HSC ne sont pas affectées par les mesures de la période :
 - ElapsedTime est une valeur entière double non signée en nanosecondes représentant la durée du premier événement de comptage au dernier événement de comptage dans l'intervalle de mesure. Si EdgeCount = 0, ElapsedTime est le temps qui s'est écoulé depuis le dernier événement de comptage dans un précédent intervalle. ElapsedTime a une plage comprise entre 0 et 4,294,967,280 ns (0x0000 0000 to 0xFFFF FFF0). La valeur 4,294,967,295 (0xFFFF FFFF) indique un débordement. Les valeurs comprises entre 0xFFFF FFF1 et 0xFFFF FFFE sont réservées.
 - EdgeCount est une valeur entière double non signée représentant le nombre d'événements de comptage dans l'intervalle de mesure.

Une entrée ne peut servir pour deux fonctions différentes, mais toute entrée qui n'est pas utilisée pour le mode activé de son compteur rapide peut être utilisée à d'autres fins. Par exemple, si vous utilisez HSC1 dans un mode qui utilise des entrées intégrées mais n'utilise pas la troisième entrée de réinitialisation externe (affectation par défaut I0.3), vous pouvez utiliser I0.3 pour des alarmes sur front ou pour HSC2.

Tableau 6- 30 Modes de comptage des compteurs rapides

Type	Entrée 1	Entrée 2	Entrée 3	Fonction
Compteur monophasé avec gestion interne du sens de comptage	Horloge	-	-	Comptage ou fréquence
			Réinitialisation	Comptage
Compteur monophasé avec gestion externe du sens de comptage	Horloge	Sens	-	Comptage ou fréquence
			Réinitialisation	Comptage
Compteur biphasé avec deux entrées d'horloge	Horloge, incrémentation	Horloge, décrémentation	-	Comptage ou fréquence
			Réinitialisation	Comptage
Compteur en quadrature de phase A/B	Phase A	Phase B	-	Comptage ou fréquence
			Réinitialisation ¹	Comptage

¹ Pour un codeur : Phase Z, Accueil

Adresses d'entrée pour les compteurs rapides

Lorsque vous configurez la CPU, vous avez la possibilité d'activer et de configurer les "entrées Matériel" pour chaque HSC.

Toutes les entrées HSC doivent être connectées aux terminaux sur le module de la CPU ou sur le signal board optionnel qui se branche en façade du module de la CPU.

Remarque

Comme on le voit dans les tableaux suivants, les affectations par défaut pour les signaux optionnels des différents HSC se recoupent. Par exemple, la réinitialisation externe optionnelle pour le HSC 1 utilise la même entrée que l'une des entrées pour le HSC 2.

Pour des CPU V4 ou ultérieures, vous pouvez réaffecter les entrées HSC pendant la configuration de la CPU. Vous n'avez pas besoin d'utiliser les affectations d'entrée par défaut.

Vérifiez toujours que vous avez configuré vos compteurs rapides de sorte qu'une entrée **n'est pas utilisée par deux compteurs**.

Les tableaux suivants montrent les affectations des entrées HSC par défaut pour les E/S intégrées des CPU et pour un SB optionnel. (si le modèle de SB sélectionné a uniquement 2 entrées, seules les entrées 4.0 et 4.1 sont disponibles).

Définitions du tableau d'entrée HSC

- Monophasé** : **C** est l'entrée d'horloge, **[d]** est l'entrée de direction (optionnel) et **[R]** est l'entrée de réinitialisation externe (optionnel)
(La réinitialisation est disponible uniquement pour le mode "Comptage".)
- Biphasé** : **CU** est l'entrée d'horloge haute, **CD** est l'entrée d'horloge basse et **[R]** est l'entrée de réinitialisation externe (optionnel)
(La réinitialisation est disponible uniquement pour le mode "Comptage".)
- quadrature de phase A/B** : **A** est l'entrée d'horloge A, **B** est l'entrée d'horloge B et **[R]** est une entrée de réinitialisation externe (optionnelle). (la réinitialisation est disponible uniquement pour le mode "comptage").

Tableau 6- 31CPU 1211C : Affectations d'adresse par défaut HSC

mode compteur HSC		Entrée intégrée à la CPU (par défaut 0.x)						Entrée SB optionnelle (4.x par défaut) ¹			
		0	1	2	3	4	5	0	1	2	3
HSC 1	monophasé	C	[d]		[R]			C	[d]		[R]
	Biphasé	CU	CD		[R]			CU	CD		[R]
	Phase A/B	A	B		[R]			A	B		[R]
HSC 2	monophasé		[R]	C	[d]				[R]	C	[d]
	Biphasé		[R]	CU	CD				[R]	CU	CD
	Phase A/B		[R]	A	B				[R]	A	B
HSC 3	monophasé					C	[d]	C	[d]		[R]
	Biphasé										
	Phase A/B										

mode compteur HSC		Entrée intégrée à la CPU (par défaut 0.x)						Entrée SB optionnelle (4.x par défaut) ¹			
		0	1	2	3	4	5	0	1	2	3
HSC4	monophasé					C	[d]	C	[d]		R]
	Biphasé					CU	CD				
	Phase A/B					A	B				
HSC 5	monophasé							C	[d]		[R]
	Biphasé							CU	CD		[R]
	Phase A/B							A	B		[R]
HSC 6	monophasé								[R]	C	[d]
	Biphasé								[R]	CU	CD
	Phase A/B								[R]	A	B

¹ Un SB avec seulement 2 entrées TOR comporte uniquement les entrées 4.0 et 4.1.

Tableau 6- 32CPU 1212C: Affectations d'adresse par défaut HSC

mode compteur HSC		Entrée intégrée à la CPU (par défaut 0.x)							Entrée SB optionnelle (par défaut 4.x) ¹				
		0	1	2	3	4	5	6	7	0	1	2	3
HSC 1	monophasé	C	[d]		[R]					C	[d]		[R]
	Biphasé	CU	CD		[R]					CU	CD		[R]
	Phase A/B	A	B		[R]					A	B		[R]
HSC 2	monophasé		[R]	C	[d]						[R]	C	[d]
	Biphasé		[R]	CU	CD						[R]	CU	CD
	Phase A/B		[R]	A	B						[R]	A	B
HSC 3	monophasé					C	[d]			[R]	C	[d]	
	Biphasé					CU	CD			[R]			
	Phase A/B					A	B			[R]			
HSC 4	monophasé						[R]	C	[d]	C	[d]		[R]
	Biphasé						[R]	CU	CD				
	Phase A/B						[R]	A	B				
HSC 5	monophasé									C	[d]		[R]
	Biphasé									CU	CD		[R]
	Phase A/B									A	B		[R]
HSC 6	monophasé										[R]	C	[d]
	Biphasé										[R]	CU	CD
	Phase A/B										[R]	A	B

¹ Un SB avec seulement 2 entrées TOR comporte uniquement les entrées 4.0 et 4.1.

Tableau 6- 33CPU 1214C, CPU 1215C et CPU 1217C :
affectations d'adresse par défaut HSC
(entrées intégrées uniquement, voir le tableau suivant pour les adresses SB optionnelles)

mode compteur HSC		Octet d'entrée TOR 0 (par défaut : 0.x)								Octet d'entrée TOR 1 (par défaut : 1.x)					
		0	1	2	3	4	5	6	7	0	1	2	3	4	5
HSC 1	monophasé	C	[d]		[R]										
	Biphasé	CU	CD		[R]										
	Phase A/B	A	B		[R]										
HSC 2	monophasé		[R]	C	[d]										
	Biphasé		[R]	CU	CD										
	Phase A/B		[R]	A	B										
HSC 3	monophasé					C	[d]		[R]						
	Biphasé					CU	CD		[R]						
	Phase A/B					A	B		[R]						
HSC 4	monophasé						[R]	C	[d]						
	Biphasé						[R]	CU	CD						
	Phase A/B						[R]	A	B						
HSC 5	monophasé									C	[d]	[R]			
	Biphasé									CU	CD	[R]			
	Phase A/B									A	B	[R]			
HSC 6	monophasé												C	[d]	[R]
	Biphasé												CU	CD	[R]
	Phase A/B												A	B	[R]

Tableau 6- 34 SB optionnel dans les CPU dans le tableau ci-dessus : Affectations d'adresse par défaut HSC

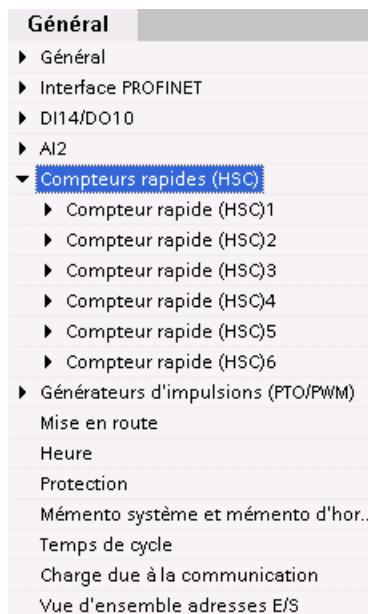
HSC		Entrées de SB optionnelles (par défaut : 4.x) ¹			
		0	1	2	3
HSC 1	Monophase	C	[d]		[R]
	Biphasé	CU	CD		[R]
	Phase A/B	A	B		[R]
HSC 2	monophasé		[R]	C	[d]
	Biphasé		[R]	CU	CD
	Phase A/B		[R]	A	B
HSC 5	Monophase	C	[d]		[R]
	Biphasé	CU	CD		[R]
	Phase A/B	A	B		[R]
HSC 6	monophasé		[R]	C	[d]
	Biphasé		[R]	CU	CD
	Phase A/B		[R]	A	B

¹ Un SB avec seulement 2 entrées TOR comporte uniquement les entrées 4.0 et 4.1.

Remarque

Les points d'E/S TOR utilisés par les compteurs rapides sont affectés pendant la configuration de l'appareil de la CPU. Lorsque des adresses d'E/S TOR sont affectées à des appareils HSC, les valeurs dans les adresses d'E/S affectées ne peuvent pas être modifiées par la fonction de forçage permanent dans une table de visualisation.

6.6.2 Configuration du HSC



Vous pouvez configurer 6 compteurs rapides au maximum. Editez la configuration d'appareil de la CPU et affectez les propriétés HSC à chaque HSC individuel. Pour activer un HSC, sélectionnez l'option "Activer" pour cet HSC.

Utilisez les instructions CTRL_HSC et/ou CTRL_HSC_EXT dans votre programme utilisateur pour contrôler le fonctionnement du HSC.



ATTENTION

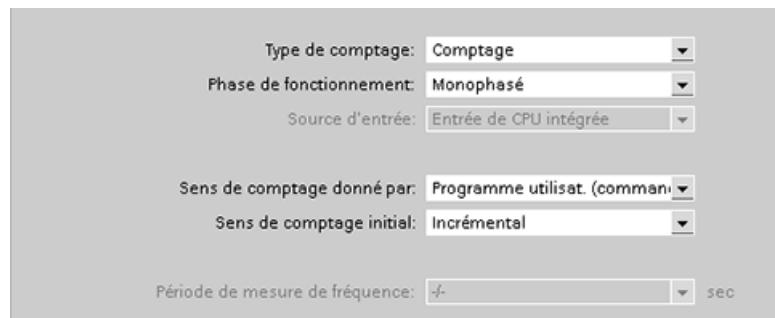
Risques liés à la modification du temps de filtre pour des voies d'entrée TOR

Si le temps de filtre pour une voie d'entrée TOR est modifié par rapport à un paramétrage précédent, une nouvelle valeur d'entrée de niveau "0" peut devoir être présente pendant une durée cumulée allant jusqu'à 20,0 ms pour que le filtre réagisse pleinement aux nouvelles entrées. Pendant ce temps, les événements d'impulsion "0" courts de moins de 20,0 ms peuvent ne pas être détectés ni comptés.

Cette modification des temps de filtre peut provoquer un fonctionnement inattendu des machines ou du processus, pouvant entraîner la mort, des blessures graves et des dommages matériels importants.

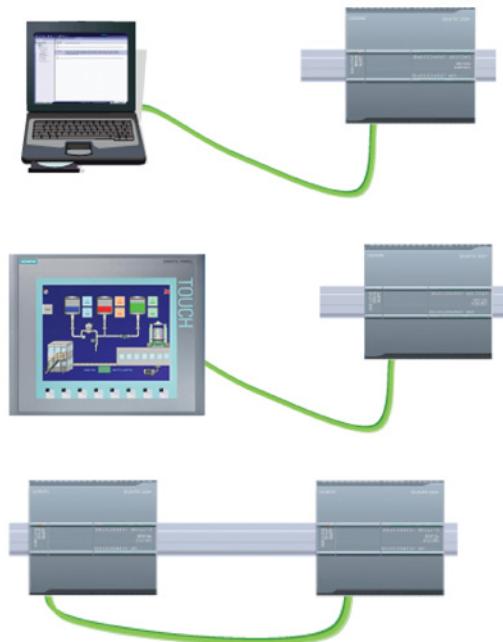
Pour garantir la prise en compte immédiate d'un nouveau temps de filtre, mettez la CPU hors tension puis sous tension.

Une fois le HSC activé, configurez les autres paramètres, tels que la fonction de comptage, les valeurs initiales, les options de réinitialisation et les événements d'alarme.



Pour plus d'informations sur la configuration du HSC, reportez-vous à la section Configuration de la CPU (Page 82).

Communication aisée entre les appareils



Pour une liaison directe entre une console de programmation et une CPU :

- Le projet doit contenir la CPU.
- La console de programmation ne fait pas partie du projet, mais doit fonctionner avec STEP 7.

Pour une liaison directe entre un tableau IHM et une CPU, le projet doit inclure à la fois la CPU et l'IHM.

Pour une liaison directe entre deux CPU :

- Le projet doit contenir les deux CPU.
- Vous devez configurer une liaison réseau entre les deux CPU.

La CPU S7-1200 est un contrôleur E/S PROFINET et communique avec STEP 7 sur une console de programmation, avec des appareils IHM et d'autres CPU ou des appareils de marque autre que Siemens. Un commutateur Ethernet n'est pas nécessaire pour la liaison directe entre une console de programmation ou IHM et une CPU. Un commutateur Ethernet est obligatoire pour un réseau comportant plus de deux CPU ou appareils IHM.

En ajoutant un CM PROFIBUS, votre CPU peut également fonctionner comme maître ou esclave sur un réseau PROFIBUS.

D'autres interfaces de communication (CM, CP ou CB) prennent en charge une variété de protocoles, par exemple Point-à-Point (PTP), Modbus, USS et GPRS (modem), CP de sécurité et CP de commande distante.

7.1 Création d'une liaison réseau

Utilisez la "Vue du réseau" de la configuration des appareils pour créer les liaisons réseau entre les appareils dans votre projet. Une fois la liaison réseau créée, servez-vous de l'onglet "Propriétés" de la fenêtre d'inspection pour configurer les paramètres du réseau.

Tableau 7- 1 Création d'une liaison réseau

Action	Résultat
Sélectionnez la "Vue du réseau" pour afficher les appareils à connecter.	
Sélectionnez le port sur un appareil et tracez la liaison vers le port sur le deuxième appareil.	
Relâchez le bouton de la souris pour créer la liaison réseau.	

7.2

Options de communication

Le S7-1200 offre plusieurs types de communication entre les CPU et les consoles de programmation, IHM et autres CPU.

ATTENTION

Si un pirate parvient à accéder physiquement aux réseaux, il peut lire et écrire les données.

TIA Portal, la CPU et les IHM (à l'exception de celles utilisant GET/PUT) utilisent une communication sécurisée qui protège contre les attaques Replay et par l'homme du milieu. Une fois que la communication est activée, l'échange des messages signés s'effectue en texte clair, ce qui permet au pirate de lire les données, mais qui protège contre l'accès aux données en écriture. C'est le TIA Portal, et non le processus de communication, qui crypte les données des blocs avec protection Know-how.

Toutes les autres formes de communication (échange I/O via PROFIBUS, PROFINET, AS-i, ou autre bus I/O, GET/PUT, blocs T et modules de communication (CM)) n'ont pas de fonction de sécurité. Vous devez les protéger en limitant leur accès physique. Si un pirate parvient à accéder physiquement aux réseaux par le biais de ces modes de communication, il peut lire et écrire les données.

Pour obtenir des informations et des recommandations sur la sécurité, voir nos "Recommandations d'opération pour la sécurité industrielle"

(http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf) sur le site de service et d'assistance de Siemens :

PROFINET

PROFINET est utilisé pour l'échange de données par le biais du programme utilisateur avec d'autres partenaires de communication via Ethernet :

- Dans le S7-1200, PROFINET prend en charge 16 périphériques ES avec 256 sous-modules au maximum, et PROFIBUS autorise 3 maîtres PROFIBUS DP indépendants, prenant en charge 32 esclaves par maître DP, avec 512 modules au maximum par maître DP.
- Communication S7
- User Datagram Protocol (UDP)
- ISO sur TCP (RFC 1006)
- Transport Control Protocol (TCP)

Contrôleur PROFINET IO

En tant qu'automate d'E/S utilisant PROFINET IO, la CPU communique avec 16 périphériques PN au plus sur le réseau PN local ou via un coupleur (link) PN/PN. Reportez-vous à PROFIBUS et PROFINET International, PI (www.profinet.com) pour plus d'informations.

PROFIBUS

PROFIBUS est utilisé pour l'échange de données par le biais du programme utilisateur avec d'autres partenaires de communication via le réseau PROFIBUS :

- Avec le CM 1242-5, la CPU fonctionne en tant qu'esclave PROFIBUS DP.
- Avec le CM 1243-5, la CPU fonctionne en tant que maître PROFIBUS DP de classe 1.
- Les esclaves PROFIBUS DP, maîtres PROFIBUS DP et interfaces AS-i (les trois modules de communication du côté gauche) et PROFINET sont des réseaux de communications indépendants qui ne se limitent pas les uns les autres.

AS-i

Le module CM 1243-2 maître AS-i S7-1200 permet le raccordement d'un réseau interface AC à une CPU S7-1200.

Communication CPU à CPU S7

Vous pouvez créer une connexion de communication sur une station partenaire et utiliser les instructions GET et PUT pour communiquer avec des CPU S7.

Communication TeleService

Avec TeleService via GPRS, une station d'ingénierie sur laquelle STEP 7 est installé communique via le réseau GSM et Internet avec une station SIMATIC S7-1200 munie d'un CP 1242-7. La liaison passe par un serveur TeleControl qui sert d'intermédiaire et qui est connecté à Internet.

IO-Link

Le maître S7-1200 SM 1278 4xIO-Link active des périphériques IO-Link pour se connecter à un S7-1200 CPU.

7.3 Liaisons de communication asynchrones V4.1

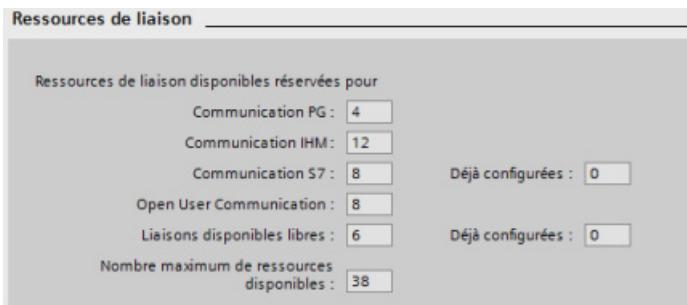
Vue d'ensemble des services de communication

La CPU accepte les services de communication suivants :

Service de communication	Fonctionnalité	Utilisation de PROFIBUS DP		Utilisation d'Ethernet
		Module maître DP CM 1243-5	Module esclave DP CM 1242-5	
Communication PG	Mise en service, test, diagnostic	Oui	Non	Oui
Communication HMI	Contrôle et surveillance par l'opérateur	Oui	Non	Oui
Communication S7	Échange de données à l'aide de connexions configurées	Oui	Non	Oui
Routage de fonctions PG	Par exemple, test et diagnostic au-delà des limites du réseau	Non	Non	Non
PROFIBUS DP	Échange de données entre maître et esclave	Oui	Oui	Non
PROFINET IO	Échange de données entre contrôleurs d'E/S et dispositifs d'E/S	Non	Non	Oui
Serveur Web	Diagnostic	Non	Non	Oui
SNMP (Protocole simple de gestion de réseau)	Protocole standard de diagnostic et de paramétrage du réseau	Non	Non	Oui
Communication ouverte via TCP/IP	Échange de données via Industrial Ethernet avec protocole TCP/IP (avec FB chargeables)	Non	Non	Oui
Communication ouverte via ISO sur TCP	Échange de données via Industrial Ethernet avec ISO sur protocole TCP (avec FB chargeables)	Non	Non	Oui
Communication ouverte via UDP	Échange de données via Industrial Ethernet avec protocole UDP (avec FB chargeables)	Non	Non	Oui

Connexions disponibles

La CPU prend en charge le nombre suivant de liaisons de communication asynchrones simultanées maximum pour PROFINET et PROFIBUS. Le nombre maximum de ressources de connexion affectées à chaque catégorie est fixe ; vous ne pouvez pas modifier ces valeurs. Cependant vous pouvez configurer les 6 "Connexions libres disponibles" pour augmenter le nombre de n'importe quelle catégorie comme requis par votre application.



En fonction des ressources de connexion affectées, le nombre suivant de connexions par dispositif est disponible :

	Console de programmation (PG)	Interface homme-machine (IHM)	Client/serveur GET/PUT	Communications ouvertes (Open User Communications)	Navigateur web
Nombre maximum de ressources de connexion	3 (prise en charge d'1 dispositif PG garantie)	12 (prise en charge de 4 dispositifs IHM garantie)	8	8	30 (prise en charge de 3 navigateurs web garantie)

À titre d'exemple, un PG a 3 ressources de connexion disponibles. Selon les fonctions PG actuellement utilisées, le PG peut utiliser en réalité 1, 2 ou 3 de ses ressources de connexion disponibles. Dans le S7-1200, vous avez l'assurance d'avoir toujours au moins une PG ; toutefois, il n'est pas possible d'en avoir plus d'une.

Le nombre d'IHM représente un autre exemple, tel qu'illustré à la figure ci-dessous. Les IHM ont 12 ressources de connexion disponibles. Selon le type ou le modèle d'IHM que vous possédez et les fonctions d'IHM que vous utilisez, chaque IHM peut utiliser en réalité 1, 2 ou 3 de ses ressources de connexion disponibles. Compte tenu du nombre de ressources de connexion disponibles utilisées, il peut être possible d'utiliser plus de 4 IHM en même temps. Cependant, la prise en charge d'au moins 4 IHM est toujours garantie. Une IHM peut utiliser ses ressources de connexion disponibles (1 chacune pour un total de 3) pour les fonctions suivantes :

- Lecture
- Ecriture
- Déclenchement d'alarme plus diagnostic

Exemple	IHM 1	IHM 2	IHM 3	IHM 4	IHM 5	Ressources de connexion disponibles totales
Ressources de connexion utilisées	2	2	2	3	3	12

Remarque

Connexions (HTTP) du serveur Web : La CPU fournit des liaisons pour plusieurs navigateurs web. Le nombre de navigateurs que la CPU peut prendre en charge simultanément dépend du nombre de connexions qu'un navigateur web donné demande/utilise.

Remarque

Les communications ouvertes, la liaison S7, l'IHM, la console de programmation et les liaisons de communication (HTTP) du serveur Web peuvent utiliser plusieurs ressources de connexion selon les fonctions utilisées.

7.4 Instructions PROFINET et PROFIBUS

Opérations PROFINET

Les opérations TSEND_C et TRCV_C rendent les communications PROFINET plus simples en combinant la fonctionnalité des opérations TCON et TDISCON avec l'opération TSEND ou TRCV.

- TSEND_C établit une liaison de communication TCP ou ISO-sur-TCP à une station partenaire, envoie des données et peut mettre fin à la liaison. Une fois que la liaison est créée et mise en place, elle est automatiquement maintenue et contrôlée par la CPU. TSEND_C rassemble les fonctions des opérations TCON, TDISCON et TSEND en une opération.
- TRCV_C établit une liaison de communication TCP ou ISO-sur-TCP à une station partenaire, envoie des données et peut mettre fin à la liaison. Une fois la liaison configurée et établie, elle est automatiquement conservée et surveillée par la CPU. L'opération TRCV_C rassemble les fonctions des opérations TCON, TDISCON et TRCV en une opération.

Les opérations TCON, TDISCON, TSEND et TRCV sont également prises en charge.

Utilisez les instructions TUSEND et TURCV pour émettre ou recevoir des données via le protocole UDP. TUSEND et TURCV (tout comme TSEND, TRCV, TCON, TDISCON) fonctionnent de façon asynchrone, ce qui signifie que le traitement de la tâche s'étend sur plusieurs appels de l'instruction.

Utilisez l'opération IP_CONF pour modifier les paramètres de configuration IP à partir de votre programme utilisateur. IP_CONF fonctionne de façon asynchrone. L'exécution s'étend sur de multiples appels.

Opérations PROFIBUS

L'opération DPNRM_DG (lecture de diagnostics) lit les données de diagnostic en cours de l'esclave DP dans le format spécifié par l'EN 50 170 Volume 2, PROFIBUS.

Instructions de périphérie décentralisée pour PROFINET, PROFIBUS et AS-i

Vous pouvez utiliser les opérations suivantes avec PROFINET, PROFIBUS et GPRS.

- Utilisez les opérations RDREC (lecture d'enregistrement) et WRREC (écriture d'enregistrement) pour transférer un enregistrement de données spécifié entre un composant, par exemple un bâti central ou un composant réparti (PROFIBUS DP ou PROFINET IO).
- Utilisez l'opération RALRM (lecture d'alarme) pour lire une alarme et ses informations depuis un esclave DP ou un appareil PROFINET IO. Les informations dans les paramètres de sortie contiennent les informations de démarrage de l'OB appelé ainsi que les informations de la source d'alarme.
- Utilisez les instructions DPRD_DAT (lecture de données cohérentes) et DPWR_DAT (écriture de données cohérentes) pour transférer des zones de données cohérentes de plus de 64 octets depuis ou vers un esclave norme DP.
- Pour PROFIBUS uniquement, utilisez l'instruction DPNRM_DG pour lire les données de diagnostic en cours d'un esclave DP dans le format spécifié par EN 50 170 Volume 2, PROFIBUS.

7.5 PROFINET

7.5.1 Communication ouverte (Open User Communication)

Le port PROFINET intégré de la CPU prend en charge plusieurs normes de communication via un réseau Ethernet :

- Transport Control Protocol (TCP)
- ISO sur TCP (RFC 1006)
- User Datagram Protocol (UDP)

Tableau 7- 2 Protocoles et leurs instructions de communication respectives

Protocole	Exemples d'utilisation	Entrée des données dans la zone de réception	Instructions de communication	Type d'adressage
TCP	Communication CPU à CPU Transport de trames	Mode ad hoc	Uniquement TRCV_C et TRCV (instructions d'héritage V4.1)	Affecte des numéros de port aux appareils local (actif) et partenaire (passif).
		Réception de données avec longueur indiquée	TSEND_C, TRCV_C, TCON, TDISCON, TSEND et TRCV (V4.1 et instructions d'héritage)	
ISO sur TCP	Communication CPU à CPU Fragmentation et réassemblage des messages	Mode ad hoc	Uniquement TRCV_C et TRCV (instructions d'héritage V4.1)	Affecte des TSAP aux appareils local (actif) et partenaire (passif).
		Gestion par le protocole	TSEND_C, TRCV_C, TCON, TDISCON, TSEND et TRCV (V4.1 et instructions d'héritage)	
UDP	Communication CPU à CPU Communication du programme utilisateur	User Datagram Protocol (protocole de datagramme utilisateur)	TUSEND et TURCV	Affecte des numéros de port aux appareils local (actif) et partenaire (passif), mais ce n'est pas une liaison spécialisée.
Communication S7	Communication CPU à CPU Lecture/écriture de données depuis/vers une CPU	Emission et réception de données avec longueur indiquée	GET et PUT	Affecte des TSAP aux appareils local (actif) et partenaire (passif).
PROFINET IO	Communication CPU à périphérique PROFINET IO	Emission et réception de données avec longueur indiquée	Intégrées	Intégré

7.5.1.1 Mode ad hoc

Typiquement, TCP et ISO sur TCP reçoivent des paquets de données d'une longueur indiquée, comprise entre 1 et 8192 octets. Toutefois, les instructions de communication TRCV_C et TRCV fournissent aussi un mode de communication "ad hoc" qui permet de recevoir des paquets de données de longueur variable allant de 1 à 1472 octets.

Remarque

Si vous stockez les données dans un DB "optimisé" (symbolique uniquement), vous ne pouvez recevoir les données qu'en tableaux de types de données Byte, Char, USInt et SInt.

Pour activer le mode ad hoc de l'instruction TRCV_C ou TRCV, activez le paramètre d'entrée ADHOC de l'instruction.

Si vous nappelez pas souvent l'instruction TRCV_C ou TRCV en mode ad hoc, vous recevrez peut-être plus d'un paquet lors d'un appel. Par exemple : Si vous deviez recevoir cinq paquets de 100 octets avec un appel, TCP transmettrait ces cinq paquets en un paquet de 500 octets alors que ISO-on-TCP restructurerait les paquets en cinq paquets de 100 octets.

7.5.1.2 ID de liaison pour les instructions Open User Communication

Lorsque vous insérez les instructions PROFINET TSEND_C, TRCV_C ou TCON dans votre programme utilisateur, STEP 7 crée un DB d'instance pour configurer la voie de communication (ou liaison) entre les appareils. Servez-vous des "Propriétés" (Page 161) de l'instruction pour configurer les paramètres de la liaison. L'ID de liaison pour cette liaison figure parmi ces paramètres.

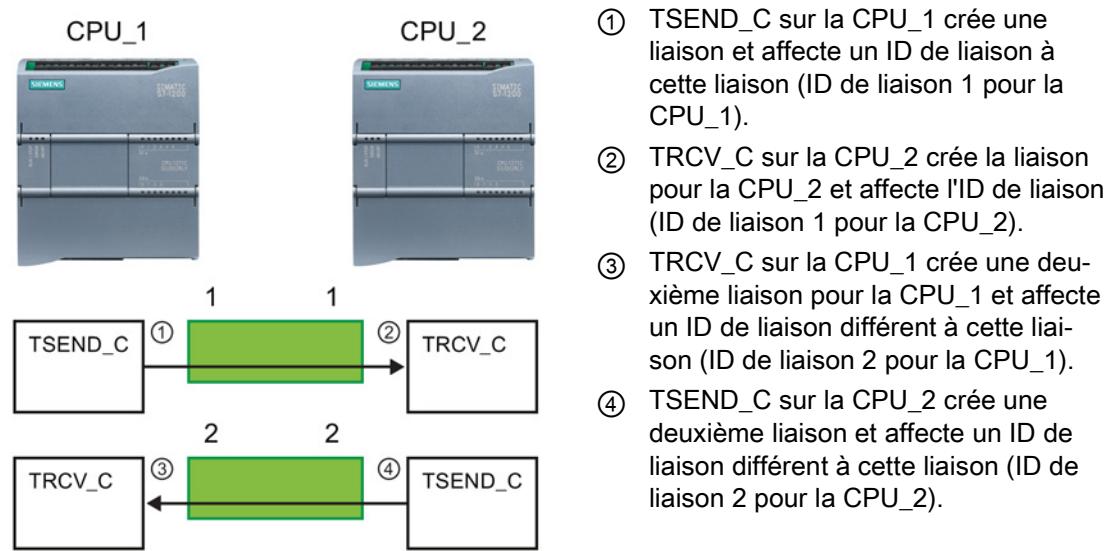
- L'ID de liaison doit être unique pour la CPU. Chaque liaison que vous créez doit avoir un DB et un ID de liaison différents.
- La CPU locale et la CPU partenaire peuvent toutes deux utiliser le même numéro d'ID de liaison pour la même liaison, mais les numéros d'ID de liaison ne doivent pas obligatoirement coïncider. Le numéro d'ID de liaison n'est significatif que pour les instructions PROFINET à l'intérieur du programme utilisateur de la CPU concernée.
- Vous pouvez utiliser n'importe quel nombre pour l'ID de liaison de la CPU. Toutefois, configurer les ID de liaison séquentiellement à partir de "1" constitue une méthode facile pour garder une trace du nombre de liaisons utilisées pour une CPU spécifique.

Remarque

Chaque instruction TSEND_C, TRCV_C ou TCON dans votre programme utilisateur crée une nouvelle liaison. Il est important d'utiliser l'ID de liaison correct pour chaque liaison.

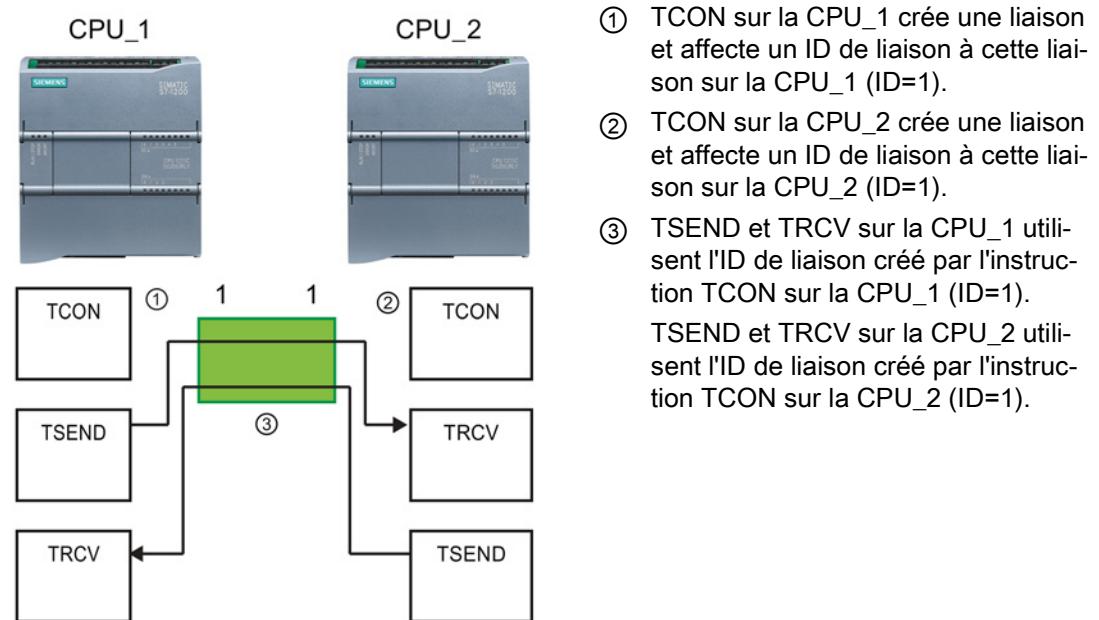
L'exemple suivant montre la communication entre deux CPU qui utilisent deux liaisons distinctes pour émettre et recevoir des données.

- L'instruction TSEND_C dans la CPU_1 se connecte à l'instruction TRCV_C dans la CPU_2 par le biais de la première liaison ("ID de liaison 1" à la fois sur la CPU_1 et la CPU_2).
- L'instruction TRCV_C dans la CPU_1 se connecte à l'instruction TSEND_C dans la CPU_2 par le biais de la deuxième liaison ("ID de liaison 2" à la fois sur la CPU_1 et la CPU_2).

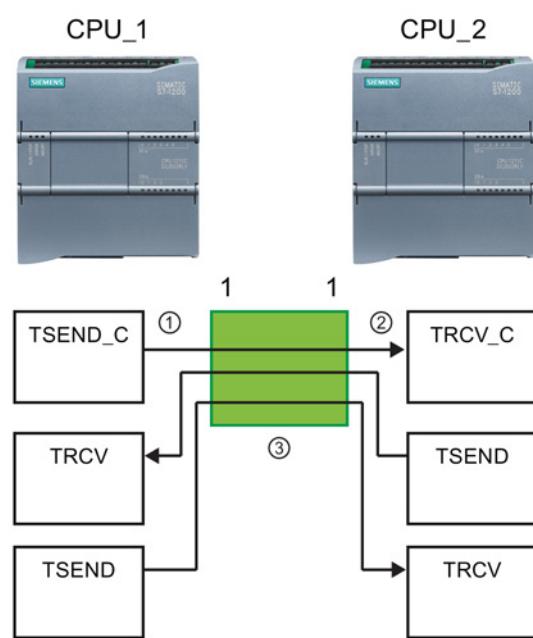


L'exemple suivant montre la communication entre deux CPU qui utilisent une même liaison pour émettre et recevoir des données.

- Chaque CPU utilise une instruction TCON pour configurer la liaison entre les deux CPU.
- L'instruction TSEND dans la CPU_1 se connecte à l'instruction TRCV dans la CPU_2 par le biais de l'ID de liaison ("ID de liaison 1") qui a été configurée par l'instruction TCON dans la CPU_1. L'instruction TRCV dans la CPU_2 se connecte à l'instruction TSEND dans la CPU_1 par le biais de l'ID de liaison ("ID de liaison 1") qui a été configurée par l'instruction TCON dans la CPU_2.
- L'instruction TSEND dans la CPU_2 se connecte à l'instruction TRCV dans la CPU_1 par le biais de l'ID de liaison ("ID de liaison 1") qui a été configurée par l'instruction TCON dans la CPU_2. L'instruction TRCV dans la CPU_1 se connecte à l'instruction TSEND dans la CPU_2 par le biais de l'ID de liaison ("ID de liaison 1") qui a été configurée par l'instruction TCON dans la CPU_1.



Comme le montre l'exemple suivant, vous pouvez également utiliser des instructions TSEND et TRCV individuelles pour communiquer par le biais d'une liaison créée par une instruction TSEND_C ou TRCV_C. Les instructions TSEND et TRCV ne créent pas elles-mêmes de nouvelle liaison de sorte qu'elles doivent utiliser le DB et l'ID de liaison créés par une instruction TSEND_C, TRCV_C ou TCON.



- ① TSEND_C sur la CPU_1 crée une liaison et affecte un ID de liaison à cette liaison (ID=1).
- ② TRCV_C sur la CPU_2 crée une liaison et affecte un ID de liaison à cette liaison sur la CPU_2 (ID=1).
- ③ TSEND et TRCV sur la CPU_1 utilisent l'ID de liaison créé par l'instruction TSEND_C sur la CPU_1 (ID=1). TSEND et TRCV sur la CPU_2 utilisent l'ID de liaison créé par l'instruction TRCV_C sur la CPU_2 (ID=1).

7.5.1.3 Paramètres pour la liaison PROFINET

Les instructions TSEND_C, TRCV_C et TCON nécessitent l'indication de paramètres relatifs à la liaison pour que la liaison au partenaire soit établie. Ces paramètres sont affectés par la structure TCON_Param pour les protocoles TCP, ISO sur TCP et UDP. Typiquement, vous utilisez l'onglet "Configuration" des "Propriétés" de l'instruction pour fournir ces paramètres. Si l'onglet "Configuration" n'est pas accessible, vous devez fournir la structure TCON_Param par programme.

TCON_Param

Tableau 7- 3 Structure de la description de la liaison (TCON_Param)

Octet	Paramètre et type de données	Description
0 ... 1	block_length	UInt Longueur : 64 octets (fixe)
2 ... 3	id	CONN_OUC (Word) Référence à cette liaison : Plage de valeurs : 1 (valeur par défaut) à 4095. Indiquez la valeur de ce paramètre pour l'instruction TSEND_C, TRCV_C ou TCON sous ID.
4	connection_type	USInt Type de liaison : <ul style="list-style-type: none"> • 17 : TCP (par défaut) • 18 : ISO sur TCP • 19 : UDP
5	active_est	Bool ID pour le type de liaison : <ul style="list-style-type: none"> • TCP et ISO sur TCP : <ul style="list-style-type: none"> – FAUX : liaison passive – VRAI : liaison active (valeur par défaut) • UDP : FAUX
6	local_device_id	USInt ID pour l'interface PROFINET ou Industrial Ethernet locale : 1 (valeur par défaut)
7	local_tsap_id_len	USInt Longueur du paramètre local_tsap_id utilisé, en octets ; valeurs possibles : <ul style="list-style-type: none"> • TCP : 0 (actif, valeur par défaut) ou 2 (passif) • ISO sur TCP : 2 à 16 • UDP : 2
8	rem_subnet_id_len	USInt Ce paramètre n'est pas utilisé.
9	rem_staddr_len	USInt Longueur de l'adresse du noeud d'extrémité partenaire, en octets : <ul style="list-style-type: none"> • 0 : non précisé (paramètre rem_staddr sans objet) • 4 (valeur par défaut) : adresse IP valide dans le paramètre rem_staddr (uniquement pour TCP et ISO sur TCP)
10	rem_tsap_id_len	USInt Longueur du paramètre rem_tsap_id utilisé, en octets ; valeurs possibles : <ul style="list-style-type: none"> • TCP : 0 (passif) ou 2 (actif, valeur par défaut) • ISO sur TCP : 2 à 16 • UDP : 0
11	next_staddr_len	USInt Ce paramètre n'est pas utilisé.

Octet	Paramètre et type de données	Description
12 ... 27	local_tsap_id	<p>Array [1..16] of Byte</p> <p>Elément d'adresse local de la liaison :</p> <ul style="list-style-type: none"> • TCP et ISO sur TCP : n° de port local (valeurs possibles : 1 à 49151 ; valeurs recommandées : 2000...5000) : <ul style="list-style-type: none"> – local_tsap_id[1] = octet de poids fort du numéro de port en notation hexadécimale – local_tsap_id[2] = octet de poids faible du numéro de port en notation hexadécimale – local_tsap_id[3-16] = sans objet • ISO sur TCP : ID TSAP local : <ul style="list-style-type: none"> – local_tsap_id[1] = B#16#E0 – local_tsap_id[2] = châssis et emplacement du noeud d'extrémité local (bits 0 à 4 : numéro d'emplacement, bits 5 à 7 : numéro de châssis) – local_tsap_id[3-16] = extension TSAP, facultative • UDP : Ce paramètre n'est pas utilisé. <p>Remarque : Veillez à ce que chaque valeur de local_tsap_id soit unique dans la CPU.</p>
28 ... 33	rem_subnet_id	Array [1..6] of USInt
34 ... 39	rem_staddr	<p>Array [1..6] of USInt</p> <p>TCP et ISO sur TCP uniquement : Adresse IP du noeud d'extrémité partenaire (sans objet pour les liaisons passives). L'adresse IP 192.168.002.003 est, par exemple, stockée dans les éléments suivants du tableau :</p> <p>rem_staddr[1] = 192 rem_staddr[2] = 168 rem_staddr[3] = 002 rem_staddr[4] = 003 rem_staddr[5-6]= sans objet</p>
40 ... 55	rem_tsap_id	<p>Array [1..16] of Byte</p> <p>Elément d'adresse partenaire de la liaison :</p> <ul style="list-style-type: none"> • TCP : numéro de port partenaire. Plage : 1 à 49151 ; valeurs recommandées : 2000 à 5000) : <ul style="list-style-type: none"> – rem_tsap_id[1] = octet de poids fort du numéro de port en notation hexadécimale – rem_tsap_id[2] = octet de poids faible du numéro de port en notation hexadécimale – rem_tsap_id[3-16] = sans objet • ISO sur TCP : ID TSAP partenaire : <ul style="list-style-type: none"> – rem_tsap_id[1] = B#16#E0 – rem_tsap_id[2] = châssis et emplacement du noeud d'extrémité partenaire (bits 0 à 4 : numéro d'emplacement, bits 5 à 7 : numéro de châssis) – rem_tsap_id[3-16] = extension TSAP, facultative • UDP : Ce paramètre n'est pas utilisé.
56 ... 61	next_staddr	Array [1..6] of Byte
62 ... 63	spare	Word
		Réserve : W#16#0000

7.5.2 Configuration du routage local/partenaire

Une liaison appareil local / partenaire (distant) définit une affectation logique de deux partenaires de communication pour établir des services de communication. Une liaison définit les éléments suivants :

- Partenaires de communication concernés (un actif, un passif)
- Type de liaison (par exemple, liaison API, IHM ou d'appareil)
- Routage

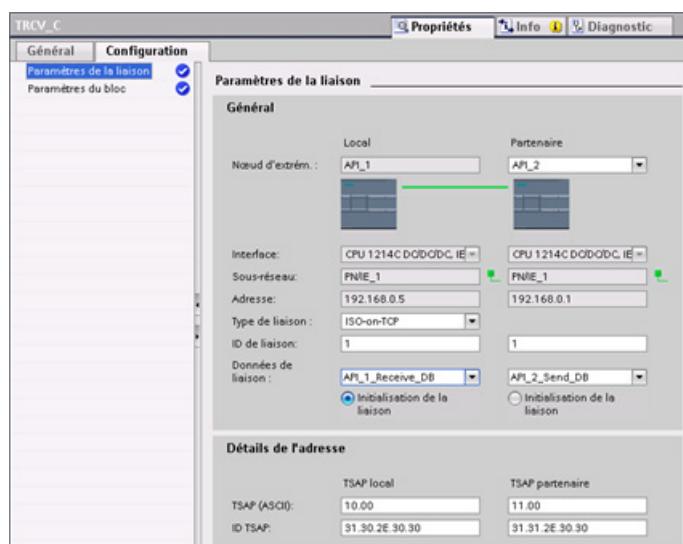
Les partenaires de communication exécutent les instructions pour configurer et établir la liaison de communication. Vous servez de paramètres pour indiquer les partenaires de nœud d'extrémité actif et passif. Une fois la liaison configurée et établie, elle est automatiquement conservée et surveillée par la CPU.

Si la liaison est interrompue (en raison d'une coupure de ligne, par exemple), le partenaire actif tente de rétablir la liaison configurée. Vous n'avez pas besoin de réexécuter l'instruction de communication.

Routages

Après insertion d'une instruction TSEND_C, TRCV_C ou TCON dans le programme utilisateur, la fenêtre d'inspection affiche les propriétés de la liaison dès que vous sélectionnez une partie quelconque de l'instruction. Indiquez les paramètres de communication dans l'onglet "Configuration" des "Propriétés" de l'instruction de communication.

Tableau 7- 4 Configuration du routage (à l'aide des propriétés de l'instruction)

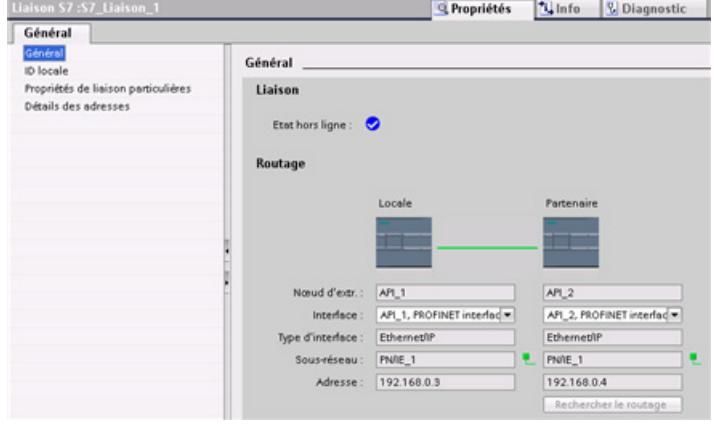
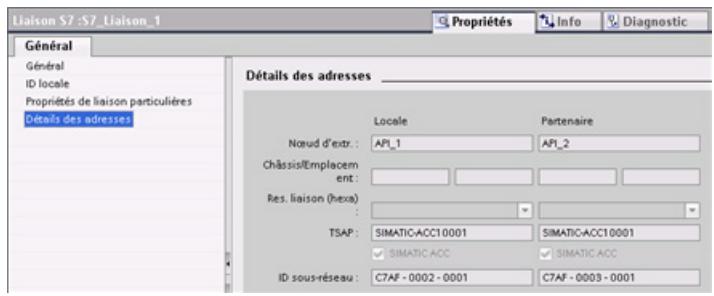
TCP, ISO sur TCP et UDP :	Propriétés de liaison
<p>Pour les protocoles Ethernet TCP, ISO sur TCP et UDP, utilisez les "Propriétés" de l'instruction (TSEND_C, TRCV_C ou TCON) pour configurer les liaisons "locale/partenaire".</p> <p>La figure montre les "Propriétés" d'une liaison ISO sur TCP dans l'onglet "Configuration".</p>	

Remarque

Lorsque vous configurez les propriétés de liaison pour une CPU, STEP 7 vous permet de sélectionner dans la CPU partenaire un DB de liaison spécifique (s'il existe) ou de créer le DB de liaison pour la CPU partenaire. La CPU partenaire doit avoir été préalablement créée pour le projet et ne peut pas être une CPU "non spécifiée".

Vous devez toujours insérer une instruction TSEND_C, TRCV_C ou TCON dans le programme utilisateur de la CPU partenaire. Lorsque vous insérez l'instruction, sélectionnez le DB de liaison qui a été créé par la configuration.

Tableau 7- 5 Configuration du routage pour la communication S7 (configuration des appareils)

Communication S7 (GET et PUT)	Propriétés de liaison
<p>Pour la communication S7, utilisez l'éditeur "Appareils & réseaux" du réseau pour configurer les liaisons locale/partenaire. Vous pouvez cliquer sur le bouton "Mis en valeur : Liaison" pour accéder aux "Propriétés".</p> <p>L'onglet "Général" fournit plusieurs propriétés :</p> <ul style="list-style-type: none"> • "Général" (montré) • "ID locale" • "Propriétés de liaison particulières" • "Détails de l'adresse" (montré) 	 

Reportez-vous à "Protocoles" (Page 156) dans le chapitre "PROFINET" ou à "Création d'une liaison S7" (Page 178) dans le chapitre "Communication S7" pour obtenir plus d'informations ainsi que la liste des instructions de communication disponibles.

Tableau 7- 6 Paramètres pour la liaison entre CPU

Paramètre	Définition	
Adresse	Adresses IP affectées	
Général	Noeud d'extrémité	Nom affecté à la CPU partenaire (réceptrice)
	Interface	Nom affecté aux interfaces
	Sous-réseau	Nom affecté aux sous-réseaux
	Type d'interface	<i>Communication S7 uniquement</i> : Type de l'interface
	Type de liaison	Type de protocole Ethernet
	ID de liaison	Numéro d'identification
	Données de liaison	Emplacement de stockage des données des CPU locale et partenaire
	Etablissement d'une liaison active	Bouton radio pour sélectionner la CPU locale ou partenaire en tant que liaison active
Détails de l'adresse	Noeud d'extrémité	<i>Communication S7 uniquement</i> : Nom affecté à la CPU partenaire (réceptrice)
	Châssis/emplACEMENT	<i>Communication S7 uniquement</i> : Adresse du châssis et de l'emplacement
	Ressources de liaison	<i>Communication S7 uniquement</i> : Composant du TSAP utilisé lors de la configuration d'une liaison S7 avec une CPU S7-300 ou S7-400
	Port (décimal)	TCP et UDP : Port de la CPU partenaire en format décimal
	TSAP ¹ et ID de sous-réseau :	ISO sur TCP (RFC 1006) et communication S7 : TSAP des CPU locale et partenaire en formats ASCII et hexadécimal

¹ Lorsque vous configurez une liaison avec une CPU S7-1200 pour ISO sur TCP, utilisez uniquement des caractères ASCII dans l'extension TSAP pour les partenaires de communication passifs.

TSAP (points d'accès au service transport)

Grâce aux points d'accès au service transport (TSAP), le protocole ISO sur TCP et la communication S7 autorisent des liaisons multiples à une adresse IP unique (liaisons 64K maximum). Les TSAP identifient de manière unique ces liaisons de noeud d'extrémité à une adresse IP.

Vous définissez les points TSAP à utiliser dans la section "Détails de l'adresse" de la boîte de dialogue "Paramètres de liaison". Vous indiquez le point TSAP d'une liaison dans la CPU dans la zone "TSAP local". Vous indiquez le point TSAP affecté à la liaison dans votre CPU partenaire dans la zone "TSAP partenaire".

Numéros de port

Avec les protocoles TCP et UDP, la configuration des paramètres de liaison de la CPU locale (active) doit indiquer l'adresse IP distante et le numéro de port de la CPU partenaire (passive).

Vous définissez les ports à utiliser dans la section "Détails de l'adresse" de la boîte de dialogue "Paramètres de liaison". Vous indiquez le port d'une liaison dans la CPU dans la zone "Port local". Vous indiquez le port affecté à la liaison dans votre CPU partenaire dans la zone "Port partenaire".

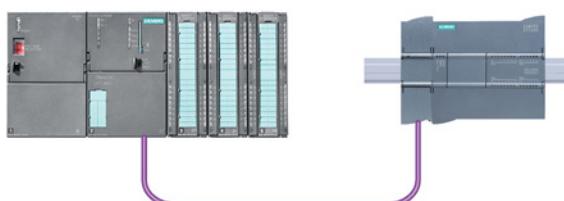
7.6 PROFIBUS

Un système PROFIBUS utilise un maître du bus pour interroger des esclaves répartis en mode multipoint sur un bus série RS485. Un esclave PROFIBUS est n'importe quel périphérique (transducteur d'E/S, vanne, entraînement de moteur ou autre appareil de mesure) qui traite des informations et envoie ses sorties au maître. L'esclave constitue une station passive sur le réseau puisqu'il n'a pas de droits d'accès au bus et peut seulement accuser réception des messages reçus ou envoyer des messages sur demande du maître. Tous les esclaves PROFIBUS ont la même priorité et le maître est à l'origine de toutes les communications de réseau.

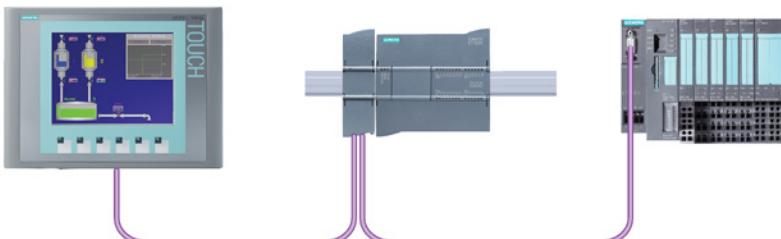
Un maître PROFIBUS constitue une station active sur le réseau. PROFIBUS DP définit deux classes de maîtres. Un maître de classe 1 (normalement un automate programmable industriel API ou un PC exécutant un logiciel spécial) gère la communication normale ou l'échange de données avec les esclaves qui lui sont affectés. Un maître de classe 2 (généralement un appareil de configuration, tel qu'un ordinateur portatif ou une console de programmation servant à la mise en service, à la maintenance ou au diagnostic) est un appareil spécial utilisé avant tout pour la mise en service des esclaves et à des fins de diagnostic.

Le S7-1200 est raccordé à un réseau PROFIBUS en tant qu'esclave DP avec le module de communication CM 1242-5. Le module CM 1242-5 (esclave DP) peut être le partenaire de communication de maîtres DP V0/V1. Si vous voulez configurer le module dans un système tiers, il y a un fichier GSD disponible pour le CM 1242-5 (esclave DP) sur le CD qui est envoyé avec le module et sur les pages de l'Assistance client Siemens Automatisation (<http://support.automation.siemens.com/WW/lisapi.dll?func=cslib.csinfo&lang=en&objid=6GK72425DX300XE0&caller=view>) sur Internet.

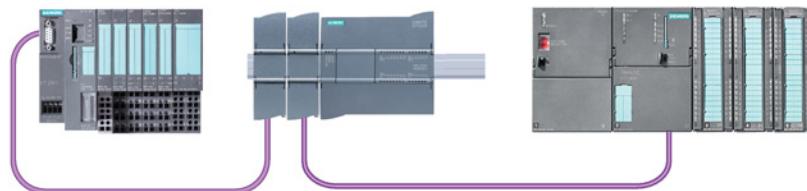
Dans la figure ci-dessous, le S7-1200 est un esclave DP affecté à un automate S7-300 :



Le S7-1200 est raccordé à un réseau PROFIBUS en tant que maître DP avec le module de communication CM 1243-5. Le module CM 1243-5 (maître DP) peut être le partenaire de communication d'esclaves DP V0/V1. Dans la figure ci-dessous, le S7-1200 est un maître pilotant un esclave DP ET200S :



Si l'on a installé à la fois un CM 1242-5 et un CM 1243-5, un S7-1200 peut opérer simultanément comme esclave d'un réseau maître DP de niveau supérieur et comme maître d'un réseau maître DP de niveau inférieur :



Dans V4.0, vous pouvez configurer un maximum de trois CM PROFIBUS par station, toute combinaison de CM maîtres ou esclaves DP étant possible. Dans une implémentation du firmware V3.0 ou ultérieur de la CPU, les maîtres DP peuvent piloter chacun 32 esclaves au maximum.

Les données de configuration des CM PROFIBUS sont stockées sur la CPU locale. Cela permet un simple remplacement de ces modules de communication si nécessaire.

7.6.1 Services de communication des CM PROFIBUS

Les CM PROFIBUS utilisent le protocole PROFIBUS DP-V1.

Types de communication avec DP-V1

Les types de communication suivants sont disponibles avec DP-V1 :

- Communication cyclique (CM 1242-5 et CM 1243-5)

Les deux modules PROFIBUS prennent en charge la communication cyclique pour le transfert de données du processus entre esclave DP et maître DP.

La communication cyclique est gérée par le système d'exploitation de la CPU. Elle ne nécessite pas de blocs logiciels. Les données d'E/S sont lues ou écrites directement dans la mémoire image de la CPU.

- Communication acyclique (CM 1243-5 uniquement)

Le module maître DP prend également en charge la communication acyclique à l'aide de blocs logiciels :

- L'instruction "RALRM" est disponible pour la gestion des alarmes.
- Les instructions "RDREC" et "WRREC" permettent de transférer les données de configuration et de diagnostic.

Fonctions non prises en charge par le CM 1243-5 : SYNC/FREEZE et Get_Master_Diag

Autres services de communication du CM 1243-5

Le module maître CM 1243-5 DP prend en charge les services de communication supplémentaires suivants :

- Communication S7

- Services PUT/GET

Le maître DP fonctionne en tant que client et serveur pour les requêtes provenant d'autres automates S7 ou de PC via PROFIBUS.

- Communication PG/OP

Les fonctions PG permettent de charger les données de configuration et les programmes utilisateur d'une console de programmation dans la CPU et de transférer les données de diagnostic dans une console de programmation.

Les partenaires possibles pour la communication OP sont les pupitres HMI Panels, les SIMATIC Panel PC avec WinCC flexible ou les systèmes SCADA qui supportent la communication S7.

7.6.2

Référence aux manuels utilisateurs des CM PROFIBUS

Informations supplémentaires

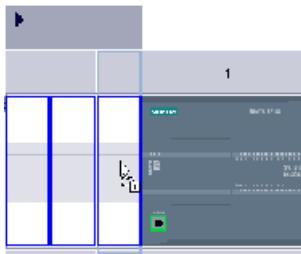
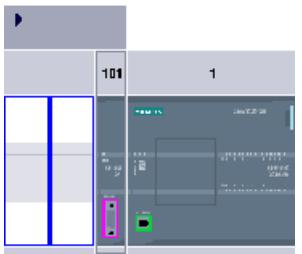
Vous trouverez des informations détaillées sur les modules de communication PROFIBUS dans les manuels de ces appareils. Ces manuels sont disponibles sur Internet, dans les pages du service client de Siemens Industrial Automation sous les entrées suivantes :

- CM 1242-5 (<http://support.automation.siemens.com/WW/view/en/49852105>)
- CM 1243-5 (<http://support.automation.siemens.com/WW/view/en/49851842>)

7.6.3 Ajout du module CM 1243-5 (maître DP) et d'un esclave DP

Dans le portail "Appareils & Réseaux", utilisez le catalogue du matériel pour ajouter des modules PROFIBUS à la CPU. Ces modules se raccordent sur le côté gauche de la CPU. Pour insérer un module dans la configuration matérielle, sélectionnez le module concerné dans le catalogue du matériel et double-cliquez ou faites glisser le module dans l'emplacement mis en évidence.

Tableau 7- 7 Ajout d'un module CM 1243-5 PROFIBUS (maître DP) à la configuration des appareils

Module	Sélectionnez le module	Insérez le module	Résultat
CM 1243-5 (maître DP)			

Utilisez également le catalogue du matériel pour ajouter des esclaves DP. Par exemple, pour ajouter un esclave DP ET200 S, affichez le détail des conteneurs suivants dans le Catalogue du matériel :

- Périphérie décentralisée
- ET200 S
- Coupleurs
- PROFIBUS

Puis, sélectionnez "6ES7 151-1BA02-0AB0" (IM151-1 HF) dans la liste des numéros de référence et ajoutez l'esclave DP ET200 S comme illustré dans la figure ci-dessous.

Tableau 7- 8 Ajout d'un esclave DP ET200 S à la configuration des appareils

Insérez l'esclave DP	Résultat
	

7.6.4 Affectation d'adresses PROFIBUS au module CM 1243-5 et à l'esclave DP

Configuration de l'interface PROFIBUS

Une fois que vous avez configuré les liaisons réseau logiques entre deux appareils PROFIBUS, vous pouvez configurer les paramètres des interfaces PROFIBUS. Pour ce faire, cliquez sur le carré PROFIBUS violet sur le module CM 1243-5. L'onglet "Propriétés" de la fenêtre d'inspection affiche l'interface PROFIBUS. L'interface PROFIBUS de l'esclave DP se configure de la même manière.

Tableau 7- 9 Configuration des interfaces PROFIBUS du module CM 1243-5 (maître DP) et de l'esclave DP ET200 S

Module CM 1243-5 (maître DP)	Esclave DP ET200 S

① port PROFIBUS

Affectation de l'adresse PROFIBUS

Dans un réseau PROFIBUS, une adresse PROFIBUS est affectée à chaque appareil. Cette adresse est comprise entre 0 et 127, aux exceptions suivantes près :

- Adresse 0 : Réservée aux outils de configuration et/ou de programmation du réseau raccordés au bus
- Adresse 1 : Réservée au premier maître par Siemens
- Adresse 126 : Réservée aux appareils sortant de l'usine qui n'ont pas de réglage par commutateur et doivent être réadressés via le réseau
- Adresse 127 : Réservée aux messages à diffusion générale vers tous les appareils dans le réseau et ne pouvant pas être affectée à des appareils opérationnels.

Ainsi, les adresses pouvant être utilisées par des appareils PROFIBUS opérationnels vont de 2 à 125.

Dans la fenêtre Propriétés, sélectionnez l'entrée de configuration "Adresse PROFIBUS". STEP 7 affiche la boîte de dialogue de configuration de l'adresse PROFIBUS qui sert à définir l'adresse PROFIBUS de l'appareil.

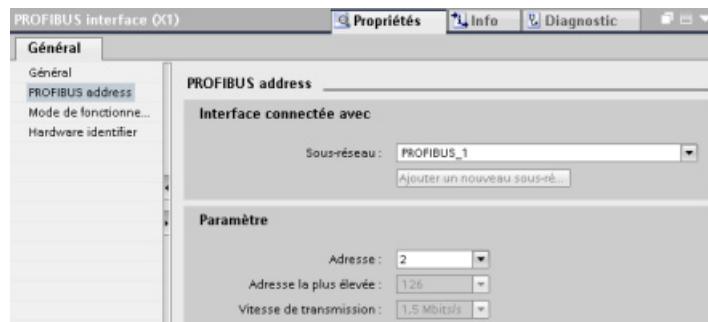


Tableau 7- 10 Paramètres pour l'adresse PROFIBUS

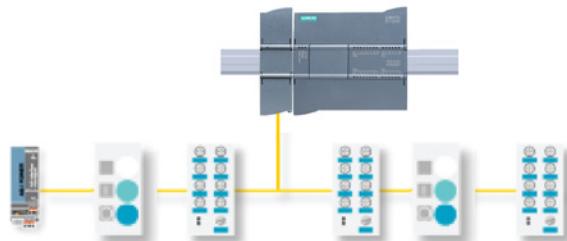
Paramètre	Description	
Sous-réseau	<p>Nom du sous-réseau auquel l'appareil est connecté. Cliquez sur le bouton "Ajouter nouveau sous-réseau" pour créer un nouveau sous-réseau. "non connecté" est la valeur par défaut. Deux types de connexion sont possibles :</p> <ul style="list-style-type: none"> • La valeur par défaut "non connecté" fournit une liaison locale. • Un sous-réseau est nécessaire lorsque votre réseau contient au moins deux appareils. 	
Paramètres	Adresse	Adresse PROFIBUS affectée à l'appareil
	Adresse la plus élevée	L'adresse PROFIBUS la plus élevée est basée sur les stations actives dans le réseau PROFIBUS (par exemple, maître DP). Les esclaves DP passifs ont, indépendamment de cela, des adresses PROFIBUS allant de 1 à 125 même si, par exemple, l'adresse PROFIBUS la plus élevée est définie à 15. L'adresse PROFIBUS la plus élevée est pertinente pour le passage du jeton (c'est-à-dire des droits d'émission), le jeton étant uniquement transmis à des stations actives. Indiquer l'adresse PROFIBUS la plus élevée permet d'optimiser le bus.
	Vitesse de transmission	Vitesse de transmission du réseau PROFIBUS configuré : Les vitesses de transmission PROFIBUS vont de 9,6 Kbits/s à 12 Mbits/s. Le réglage de la vitesse de transmission dépend des propriétés des noeuds PROFIBUS utilisés. La vitesse de transmission ne doit pas dépasser la vitesse prise en charge par le noeud le plus lent. La vitesse de transmission est normalement définie pour le maître sur le réseau PROFIBUS, tous les esclaves DP utilisant automatiquement la même vitesse de transmission (détection automatique de débit).

7.7 AS-i

Le module CM 1243-2 maître AS-i S7-1200 permet le raccordement d'un réseau AS-i à une CPU S7-1200.

L'interface actionneur/capteur, interface AS-i, est un réseau d'interconnexion à un seul maître pour le niveau le plus bas dans les systèmes d'automatisation. Le module CM 1243-2 sert de maître AS-i pour le réseau. Avec un câble AS-i unique, il est possible de connecter les capteurs et actionneurs (esclaves AS-i) à la CPU par le biais du CM 1243-2. Le CM 1243-2 gère toute la coordination du réseau AS-i et transmet les données et informations d'état des actionneurs et capteurs à la CPU par le biais des adresses d'E/S qui lui sont affectées. Vous pouvez accéder aux valeurs binaires ou analogiques en fonction du type d'esclave. Les esclaves AS-i sont les voies d'entrée et de sortie du réseau AS-i et ne sont actifs que si le CM 1243-2 les appelle.

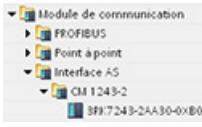
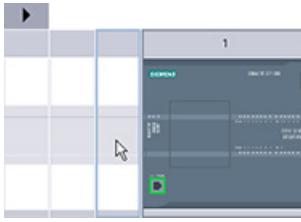
Dans la figure ci-dessous, le S7-1200 est un maître AS-i qui pilote des modules esclaves TOR/analogiques AS-i.



7.7.1 Ajout du module maître AS-i CM 1243-2 et d'un esclave AS-i

Utilisez le catalogue du matériel pour ajouter des modules maître AS-i CM 1243-2 à la CPU. Ces modules se raccordent sur le côté gauche de la CPU, trois modules maître AS-i CM 1243-2 pouvant être utilisés au maximum. Pour insérer un module dans la configuration matérielle, sélectionnez le module concerné dans le catalogue du matériel et double-cliquez ou faites glisser le module dans l'emplacement mis en évidence.

Tableau 7- 11 Ajout d'un module maître AS-i CM 1243-2 à la configuration des appareils

Module	Sélectionnez le module	Insérez le module	Résultat
CM 1243-2 maître AS-i			

Utilisez également le catalogue du matériel pour ajouter des esclaves AS-i. Par exemple, pour ajouter un esclave qui soit "module d'E/S, compact, TOR, entrées", affichez le détail des dossiers suivants dans le Catalogue du matériel :

- Appareils de terrain
- Esclaves AS Interface

Puis, sélectionnez "3RG9 001-0AA00" (AS-i SM-U, 4DI) dans la liste des numéros de référence et ajoutez l'esclave "module d'E/S, compact, TOR, entrées" comme illustré dans la figure ci-après.

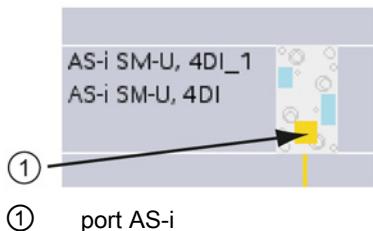
Tableau 7- 12 Ajout d'un esclave AS-i à la configuration des appareils

Insérez l'esclave AS-i	Résultat
	

7.7.2 Affectation d'une adresse AS-i à un esclave AS-i

Configuration de l'interface esclave AS-i

Pour paramétriser l'interface AS-i, cliquez sur le carré AS-i jaune sur l'esclave AS-i. L'onglet "Propriétés" de la fenêtre d'inspection affiche l'interface AS-i.



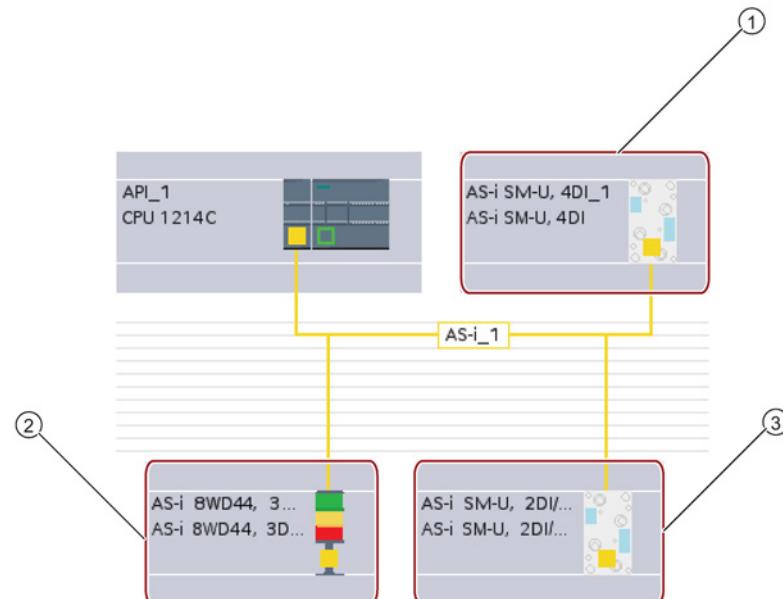
Affectation de l'adresse esclave AS-i

Dans un réseau AS-i, une adresse d'esclave AS-i est affectée à chaque appareil. Cette adresse est comprise entre 0 et 31, l'adresse 0 étant toutefois réservée aux nouveaux esclaves. Les adresses d'esclaves vont de 1(A ou B) à 31(A ou B) pour un total maximal de 62 esclaves.

Les appareils AS-i "standard" utilisent l'adresse entière et ont une adresse numérotée sans la dénomination A ou B. Les appareils AS-i "abonné A/B" utilisent la partie A ou B de chaque adresse qui permet d'utiliser deux fois chacune des 31 adresses. Les plages d'adresse vont de 1A à 31A et de 1B à 31B.

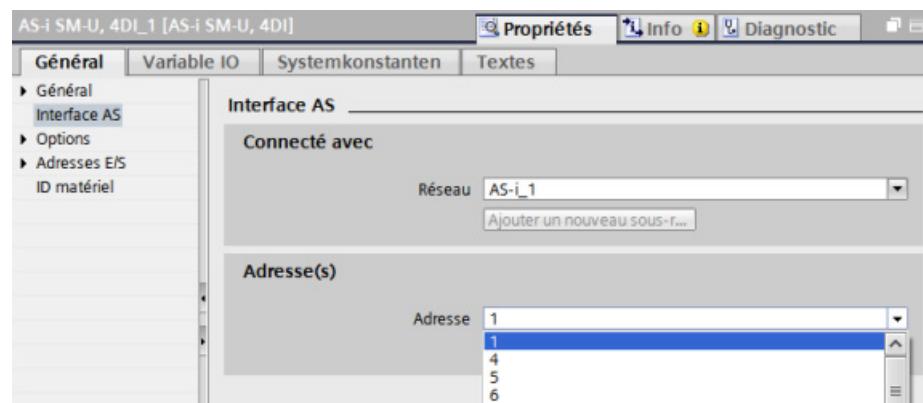
Toute adresse dans la plage de 1 à 31 peut être affectée à un esclave AS-i ; il est indifférent que les esclaves commencent à l'adresse 21 ou que le premier esclave reçoive effectivement l'adresse 1.

Dans l'exemple ci-dessous, trois appareils AS-i ont été adressés comme "1" (un appareil de type standard), "2A" (un appareil de type abonné A/B) et "3" (un appareil de type standard) :



- ① Adresse esclave AS-i 1, Appareil : SM-U AS-i, 4DI, référence : 3RG9 001-0AA00
- ② Adresse esclave AS-i 2A ; Appareil : 8WD44 AS-i, 3DO, A/B, référence : 8WD4 428-0BD
- ③ Adresse esclave AS-i 3, Appareil : SM-U AS-i, 2DI/2DO, référence : 3RG9 001-0AC00

Entrez l'adresse d'esclave AS-i ici :



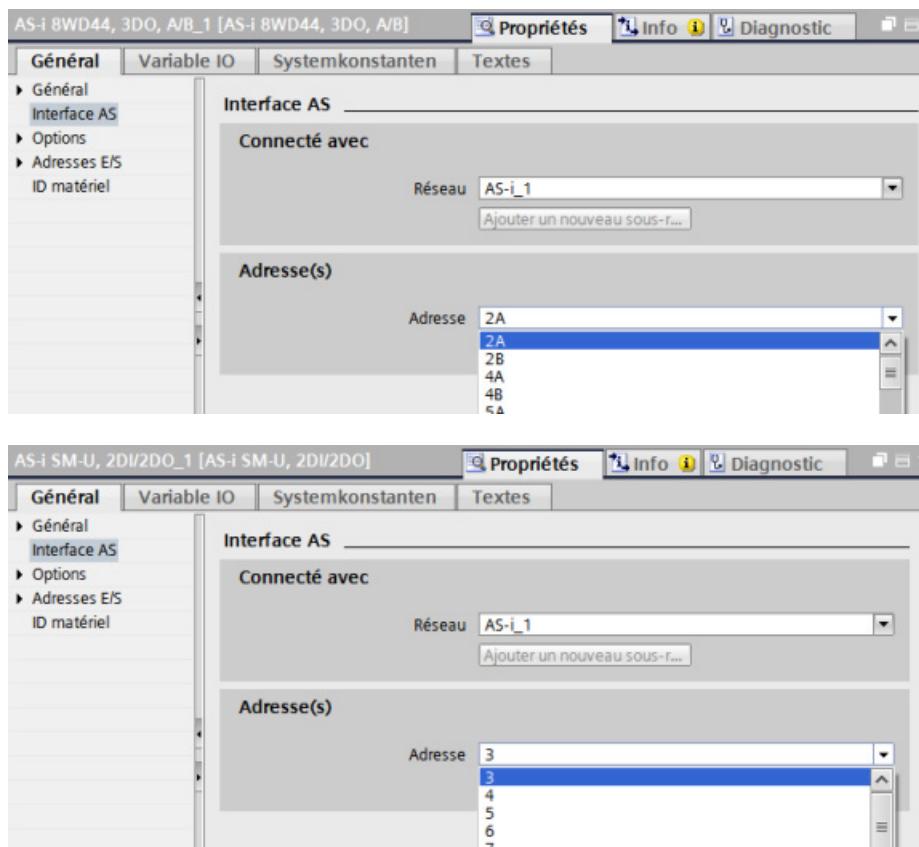


Tableau 7- 13Paramètres pour l'interface AS-i

Paramètre	Description
Réseau	Nom du réseau auquel l'appareil est connecté.
Adresse(s)	Adresse AS-i affectée à l'esclave, comprise dans la plage allant de 1(A ou B) à 31(A ou B) pour un total maximal de 62 esclaves

7.8 Communication S7

7.8.1 Instructions GET et PUT

Vous pouvez utiliser les instructions GET et PUT pour communiquer avec des CPU S7 via des liaisons PROFINET et PROFIBUS. Ceci est possible uniquement si la fonction "Autoriser accès via communication PUT/GET" est activée pour la CPU partenaire dans la propriété "Protection" des propriétés de la CPU locale :

- Accès aux données dans une CPU éloignée : Une CPU S7-1200 peut utiliser uniquement des adresses absolues dans le champ d'entrée ADDR_x pour accéder à des variables dans des CPU éloignées (S7-200/300/400/1200).
- Accès aux données dans un DB standard : Une CPU S7-1200 peut utiliser uniquement des adresses absolues dans le champ d'entrée ADDR_x pour accéder aux variables d'un DB standard dans une CPU S7 éloignée.
- Accès aux données dans un DB optimisé : Une CPU S7-1200 ne peut pas accéder aux variables d'un DB optimisé dans une CPU S7-1200 éloignée.
- Accès aux données dans une CPU locale : Une CPU S7-1200 peut utiliser soit des adresses absolues, soit des adresses symboliques comme entrées pour les champs d'entrée RD_x ou SD_x des instructions GET et PUT respectivement.

STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

Remarque

Pour garantir la cohérence des données, évaluez toujours le moment où l'opération s'est achevée (NDR = 1 pour GET ou DONE = 1 pour PUT) avant d'accéder aux données ou de déclencher une nouvelle opération de lecture ou d'écriture.

Remarque

L'opération GET/PUT du programme de CPU V4.0 n'est pas activée automatiquement

L'opération GET/PUT du programme de CPU V3.0 est activée automatiquement dans une CPU V4.0.

Cependant, une opération GET/PUT du programme de CPU V4.0 n'est pas activée automatiquement dans une CPU V4.0. Vous devez vous rendre sur "Configuration de l'appareil" de la CPU, dans l'onglet "Propriétés" de la fenêtre d'inspection puis dans la propriété "Protection" pour activer l'accès GET/PUT (Page 89).

7.8.2 Crédation d'une liaison S7

Mécanismes de connexion

Pour accéder à des partenaires de liaison éloignés avec des instructions PUT/GET, l'utilisateur doit également avoir l'autorisation d'accès.

Par défaut, l'option "Autoriser accès via communication PUT/GET" n'est pas activée. Dans ce cas, l'accès en lecture et en écriture aux données de la CPU est possible uniquement pour les liaisons de communication qui nécessitent la configuration ou la programmation à la fois de la CPU locale et du partenaire de communication. L'accès via les instructions BSEND/BRCV est possible, par exemple.

Les liaisons pour lesquelles la CPU locale est uniquement un serveur (aucune configuration/programmation de la communication avec le partenaire de communication n'existe dans la CPU locale) ne sont donc pas possibles pendant le fonctionnement de la CPU. Exemple :

- Accès PUT/GET, FETCH/WRITER ou FTP via des modules de communication
- Accès PUT/GET à partir d'autres CPU S7
- Accès IHM via la communication PUT/GET

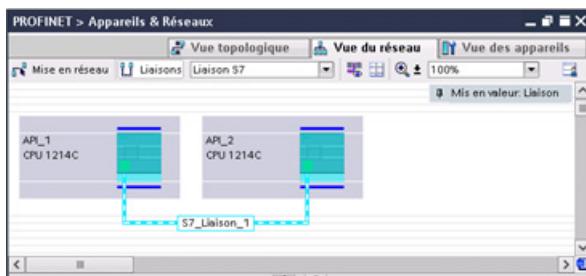
Si vous voulez autoriser l'accès aux données de la CPU à partir du client, c'est-à-dire si vous ne voulez pas restreindre les services de communication de la CPU, vous pouvez configurer la protection d'accès pour la CPU S7-1200 (Page 89) pour ce niveau de sécurité.

Types de liaisons

Le type de liaison que vous sélectionnez crée une liaison de communication vers une station partenaire. La liaison est configurée, établie et automatiquement surveillée.

Dans le portail "Appareils & Réseaux", utilisez la "Vue du réseau" pour créer les liaisons réseau entre les appareils dans votre projet. Cliquez d'abord sur l'onglet "Liaisons", puis sélectionnez le type de liaison dans la liste déroulante juste à droite (une liaison S7, par exemple). Cliquez sur le carré vert (PROFINET) du premier appareil et tracez une ligne vers le carré PROFINET du deuxième appareil. Relâchez le bouton de la souris : la liaison PROFINET est créée.

Reportez-vous à "Création d'une liaison réseau" (Page 148) pour plus d'informations.



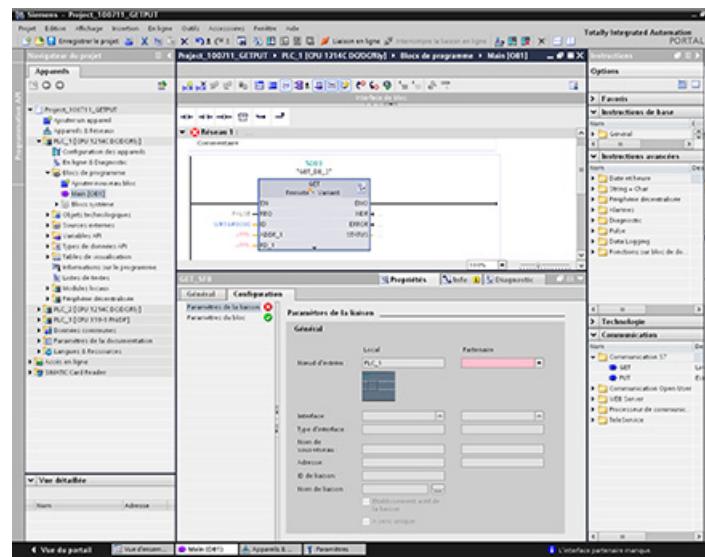
Cliquez sur le bouton "Mis en valeur : Liaison" pour accéder à la boîte de dialogue de configuration "Propriétés" de l'instruction de communication.

7.8.3

Paramétrage de la liaison GET/PUT

Le paramétrage de la liaison pour instructions GET/PUT est une aide utilisateur qui vous assiste lors de la configuration de liaisons de communication CPU à CPU S7.

Après l'insertion d'un bloc GET ou PUT, le paramétrage de la liaison pour instructions GET/PUT est démarré :



La fenêtre d'inspection affiche les propriétés de la liaison lorsque vous sélectionnez n'importe quelle partie de l'instruction. Indiquez les paramètres de communication dans l'onglet "Configuration" des "Propriétés" de l'instruction de communication.

Après l'insertion d'un bloc GET ou PUT, l'onglet "Configuration" apparaît automatiquement et la page "Paramètres de la liaison" s'affiche immédiatement. Cette page vous permet de configurer la liaison S7 nécessaire et de définir le paramètre "ID de liaison" auquel fait référence le paramètre "ID" du bloc. Une page "Paramètres du bloc" vous permet de configurer des paramètres supplémentaires du bloc.

Remarque

L'opération GET/PUT du programme de CPU V4.0 n'est pas activée automatiquement

L'opération GET/PUT du programme de CPU V3.0 est activée automatiquement dans une CPU V4.0.

Cependant, une opération GET/PUT du programme de CPU V4.0 n'est pas activée automatiquement dans une CPU V4.0. Vous devez vous rendre sur "Configuration de l'appareil" de la CPU, dans l'onglet "Propriétés" de la fenêtre d'inspection puis dans la propriété "Protection" pour activer l'accès GET/PUT (Page 89).

7.9 GPRS

7.9.1 Connexion à un réseau GSM

Communication WAN sur IP via GPRS

Le S7-1200 peut être connecté aux réseaux GSM grâce au processeur de communication CP 1242-7. Le CP 1242-7 permet une communication WAN depuis des stations distantes avec un centre de commande et une communication inter-stations.

Une communication inter-stations est possible uniquement via un réseau GSM. Pour une communication entre une station distante et une salle de commande, le centre de commande doit disposer d'un PC avec accès à Internet.

Le CP 1242-7 accepte les services de communication suivants via le réseau GSM :

- GPRS (General Packet Radio Service)

Le service orienté paquets de transmission de données "GPRS" est géré via le réseau GSM.

- (SMS) Service de messages courts

Le CP 1242-7 peut recevoir et envoyer des messages SMS. Le partenaire de communication peut être un téléphone portable ou un S7-1200.

Le CP 1242-7 est adapté pour une utilisation dans le secteur industriel partout dans le monde et accepte les bandes de fréquence suivantes :

- 850 MHz
- 900 MHz
- 1 800 MHz
- 1 900 MHz

Conditions requises

Le matériel utilisé dans les stations ou le centre de commande dépend de l'application spécifique.

- Pour une communication avec ou via une salle de commande centrale, le centre de commande doit disposer d'un PC avec un accès à Internet.
- Excepté le matériel de la station, une station S7-1200 distante avec un CP 1242-7 doit satisfaire aux exigences suivantes pour pouvoir communiquer via le réseau GSM :
 - Contrat avec un fournisseur de réseau GSM approprié

Si le système GPRS est utilisé, le contrat doit permettre l'utilisation du service GPRS.

Si une communication directe entre stations via le réseau GSM est nécessaire, le fournisseur de réseau GSM doit affecter une adresse IP fixe aux CP. Dans ce cas, la communication entre stations n'est pas réalisée via le centre de commande.

- Carte SIM appartenant au contrat
 - La carte SIM est insérée dans le CP 1242-7.
- Disponibilité locale d'un réseau GSM dans la plage de la station

7.9.2 Applications du CP 1242-7

Le CP 1242-7 peut être utilisé pour les applications suivantes :

Applications Telecontrol

- Envoi de messages par SMS

Via the CP 1242-7, la CPU d'un S7-1200 distant peut recevoir des messages SMS depuis le réseau GSM ou envoyer des messages par SMS à un téléphone portable configuré ou un S7-1200.

- Communication avec un centre de commande

Les stations S7-1200 distantes communiquent via le réseau GSM et l'Internet avec un serveur telecontrol dans la station maître. Pour le transfert de données à l'aide du système GPRS, l'application "TELECONTROL SERVER BASIC" est installée sur le serveur telecontrol dans la station maître. Le serveur telecontrol communique avec un système de commande central de niveau supérieur à l'aide de la fonction du serveur OPC intégrée.

- Communication entre stations S7-1200 via un réseau GSM

La communication entre stations distantes avec un CP 1242-7 peut être gérée de deux manières différentes :

- Communication inter-stations via une station maître

Dans cette configuration, une connexion sécurisée permanente entre stations S7-1200 qui communiquent les unes avec les autres et le serveur telecontrol est établie dans la station maître. La communication entre les stations est réalisée via le serveur telecontrol. Le CP 1242-7 fonctionne en mode "Telecontrol".

- Communication directe entre les stations

Pour une communication directe entre stations sans détournement via la station maître, des cartes SIM avec une adresse IP fixe qui permettent aux stations de s'adresser les unes aux autres directement sont utilisées. Les services de communication et fonctions de sécurité possibles (par exemple VPN) dépendent de ce qui est proposé par le fournisseur de réseau. Le CP 1242-7 fonctionne en mode "GPRS direct".

TeleService via GPRS

Une connexion TeleService peut être établie entre une station d'ingénierie avec STEP 7 et une station S7-1200 distante avec un CP 1242-7 via le réseau GSM et l'Internet. La connexion a lieu entre la station d'ingénierie via un serveur telecontrol ou une passerelle TeleService qui agit comme un intermédiaire redirigeant les trames et établissant l'autorisation. Ces PC utilisent les fonctions de l'application "TELECONTROL SERVER BASIC".

Vous pouvez utiliser la connexion TeleService pour les raisons suivantes :

- Téléchargement des données de configuration ou du programme depuis le projet STEP 7 vers la station
- Interrogation des données de diagnostic sur la station

7.9.3 Autres propriétés du CP-1242-7

Autres services et fonctions du CP 1242-7

- Synchronisation horaire du CP via l'Internet

Vous pouvez régler l'heure sur le CP de la manière suivante :

- En mode "Telecontrol", l'heure du jour est transférée par le serveur telecontrol. Le CP l'utilise pour régler sa propre heure.
- En mode "GPRS direct", le CP peut demander l'heure à l'aide du SNTP.

Pour synchroniser l'heure de la CPU, vous pouvez lire l'heure actuelle depuis la CPU à l'aide d'un bloc.

- Mise en mémoire tampon provisoire de messages à envoyer en cas de problèmes de connexion
- Disponibilité accrue grâce à la possibilité de se connecter à un serveur telecontrol de remplacement
- Volume de données optimisé (connexion temporaire)

Le CP peut être configuré dans STEP 7 avec une connexion temporaire au serveur telecontrol, ce qui constitue une alternative à une connexion permanente au serveur telecontrol. Dans ce cas, une connexion au serveur telecontrol est établie uniquement si nécessaire.

- Journalisation du volume de données

Les volumes de données transférés sont journalisés et peuvent être évalués à des fins spécifiques.

7.9.4 Configuration et connexions électriques

Configuration et remplacement de module

Pour configurer le module, l'outil de configuration suivant est requis :

STEP 7 version V11.0 SP1 ou supérieure

Pour STEP 7 V11.0 SP1, vous avez également besoin du package de prise en charge "CP 1242-7" (HSP0003001).

Pour le transfert des données de traitement à l'aide du service GPRS, utilisez les instructions de communication telecontrol dans le programme utilisateur de la station.

Les données de configuration du CP 1242-7 sont sauvegardées sur la CPU locale. Cela permet un simple remplacement du CP si nécessaire.

Vous pouvez insérer jusqu'à trois modules du type CP 1242-7 pour chaque S7-1200. Cela permet par exemple d'établir des voies de communication redondantes.

Raccordements électriques

- Alimentation électrique du CP 1242-7
Le CP possède une connexion distincte pour l'alimentation électrique 24 V CC externe.
- Interface sans fil du réseau GSM
Une antenne supplémentaire est requise pour une communication GSM. Celle-ci est reliée via la prise SMA du CP.

7.9.5 Informations supplémentaires

Informations supplémentaires

Le manuel du CP 1242-7 contient des informations détaillées. Ce manuel est disponible sur Internet, dans les pages du service client de Siemens Industrial Automation sous les entrées suivantes :

45605894 (<http://support.automation.siemens.com/WW/view/fr/45605894>)

7.9.6 Accessoires

Antenne GSM/GPRS ANT794-4MR

Les antennes suivantes sont disponibles pour une utilisation dans des réseaux GSM/GPRS et peuvent être installées à la fois en intérieur et en extérieur :

- Antenne Quadband ANT794-4MR



Nom abrégé	N° de référence	Explication
ANT794-4MR	6NH9 860-1AA00	Antenne Quadband (900, 1800/1900 MHz, UMTS) ; étanche pour les utilisations en intérieur et en extérieur ; câble de raccordement de 5 m relié de façon permanente à l'antenne ; connecteur SMA, comprenant équerre de montage, vis et prises murales

- Antenne en nappe ANT794-3M



Nom abrégé	N° de référence	Explication
ANT794-3M	6NH9 870-1AA00	Antenne en nappe (900, 1800/1900 MHz) ; étanche pour les utilisations en intérieur et en extérieur ; câble de raccordement de 1,2 m relié de façon permanente à l'antenne ; connecteur SMA, comprenant disque adhésif, montage possible avec vis

Les antennes doivent être commandées séparément.

7.9.7 Référence au manuel d'antenne GSM

Informations supplémentaires

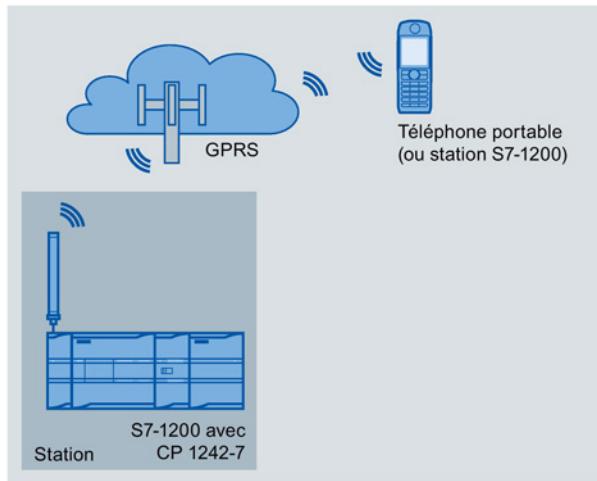
Vous trouverez des informations détaillées dans le manuel de l'appareil. Ce manuel est disponible sur Internet, dans les pages du service client de Siemens Industrial Automation sous les entrées suivantes :

23119005 (<http://support.automation.siemens.com/WW/view/fr/23119005>)

7.9.8 Exemples de configuration pour telecontrol

Vous trouverez ci-dessous des exemples de configuration pour les stations dotées d'un CP 1242-7.

Envoi de messages par SMS



Un SIMATIC S7-1200 avec un CP 1242-7 peut envoyer des messages par SMS à un téléphone portable ou une station S7-1200 configurée.

Telecontrol par un centre de commande

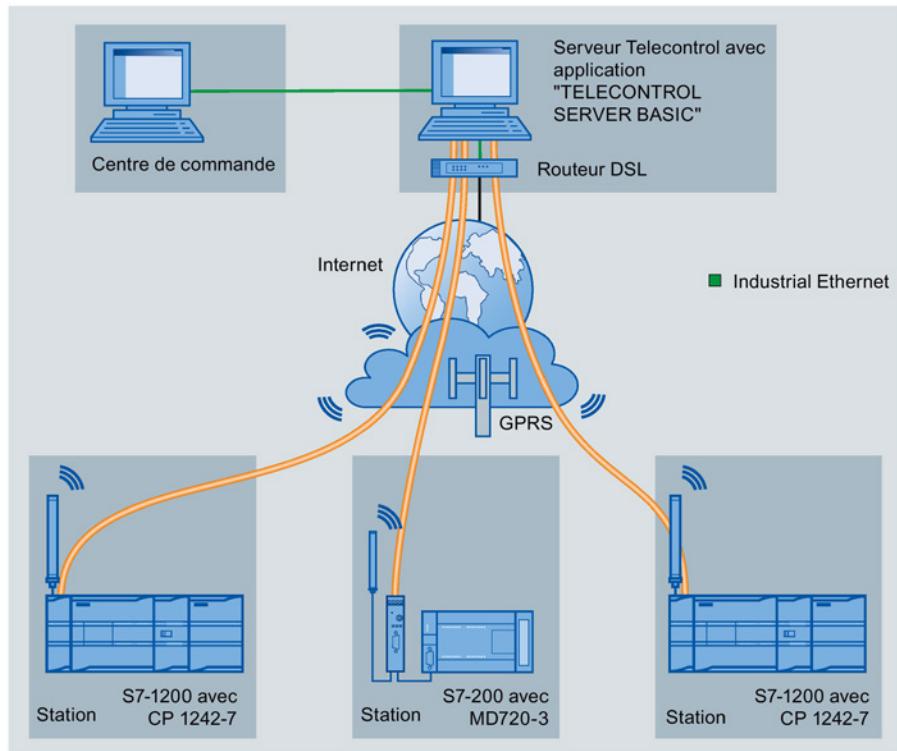


Figure 7-1 Communication entre des stations S7-1200 et un centre de commande

Dans les applications telecontrol, les stations SIMATIC S7-1200 dotées d'un CP 1242-7 communiquent avec un centre de commande via le réseau GSM et l'Internet. L'application "TELECONTROL SERVER BASIC" (TCSB) est installée sur le serveur telecontrol dans la station maître. Cela entraîne les cas d'utilisation suivants :

- Communication Telecontrol entre station et centre de commande

Dans ce cas d'utilisation, les données provenant du terrain sont envoyées par les stations au serveur telecontrol dans la station maître via le réseau GSM et Internet. Le serveur telecontrol est utilisé pour surveiller les stations distantes.

- Communication entre une station et une salle de commande avec un client OPC

Comme dans le premier cas, les stations communiquent avec le serveur telecontrol. A l'aide de son serveur OPC intégré, le serveur telecontrol échange des données avec le client OPC de la salle de commande.

Le client OPC et le serveur telecontrol peuvent se trouver sur un ordinateur unique, par exemple quand TCSB est installé sur l'ordinateur d'un centre de commande avec WinCC.

- Communication inter-stations via un centre de commande

Une communication inter-stations est possible avec des stations S7 équipées d'un CP 1242-7.

Pour permettre une communication inter-station, le serveur telecontrol transmet les messages de la station d'envoi à la station de réception.

Communication directe entre stations

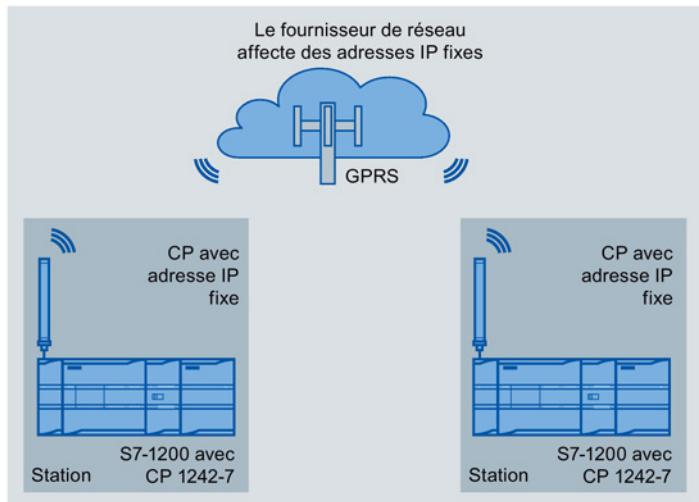


Figure 7-2 Communication directe entre deux stations S7-1200

Dans cette configuration, deux stations SIMATIC S7-1200 communiquent directement l'une avec l'autre à l'aide du CP 1242-7 via le réseau GSM. Chaque CP 1242-7 possède une adresse IP fixe. Le service correspondant du fournisseur de réseau GSM doit permettre cela.

TeleService via GPRS

Dans TeleService via GPRS, une station d'ingénierie sur laquelle STEP 7 est installé communique via le réseau GSM et l'Internet avec le CP 1242-7 dans le S7-1200.

Puisqu'un pare-feu est normalement fermé pour les demandes de connexions provenant de l'extérieur, une station de sectionnement est requise entre la station distante et la station d'ingénierie. Cette station de sectionnement peut prendre la forme d'un serveur telecontrol ou, en l'absence de serveur telecontrol dans la configuration, d'une passerelle TeleService.

TeleService avec serveur telecontrol

La connexion a lieu via le serveur telecontrol.

- La station d'ingénierie et le serveur telecontrol sont connectés via l'Intranet (LAN) ou Internet.
- Le serveur telecontrol et la station distante sont connectés via l'Internet et via le réseau GSM.

La station d'ingénierie et le serveur telecontrol peuvent également être le même ordinateur ; en d'autres termes, STEP 7 et TCSB sont installés sur le même ordinateur.

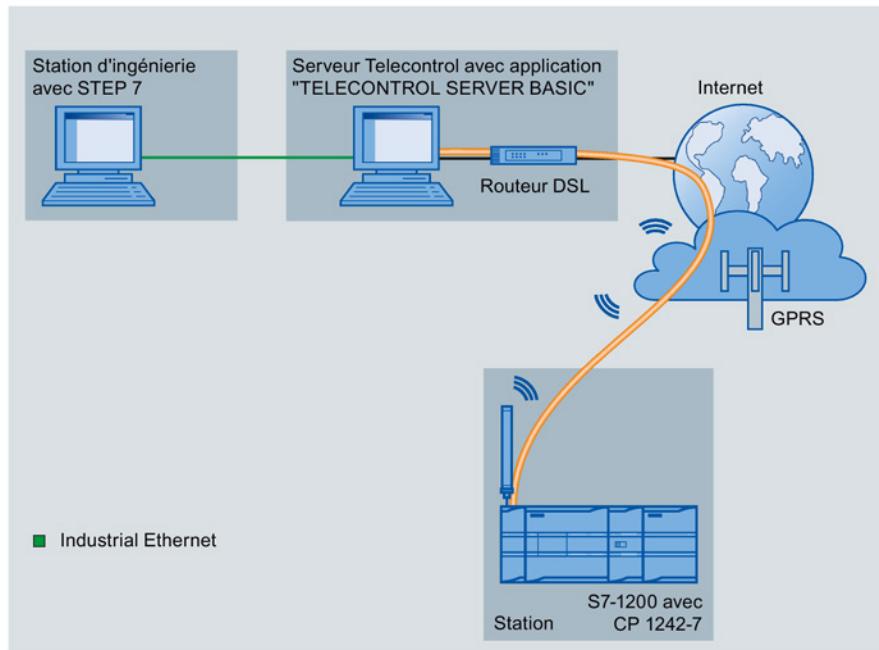


Figure 7-3 TeleService via GPRS dans une configuration avec serveur telecontrol

TeleService sans serveur telecontrol

La connexion a lieu via la passerelle TeleService.

La connexion entre la station d'ingénierie et la passerelle TeleService peut être locale via un réseau LAN ou via l'Internet.

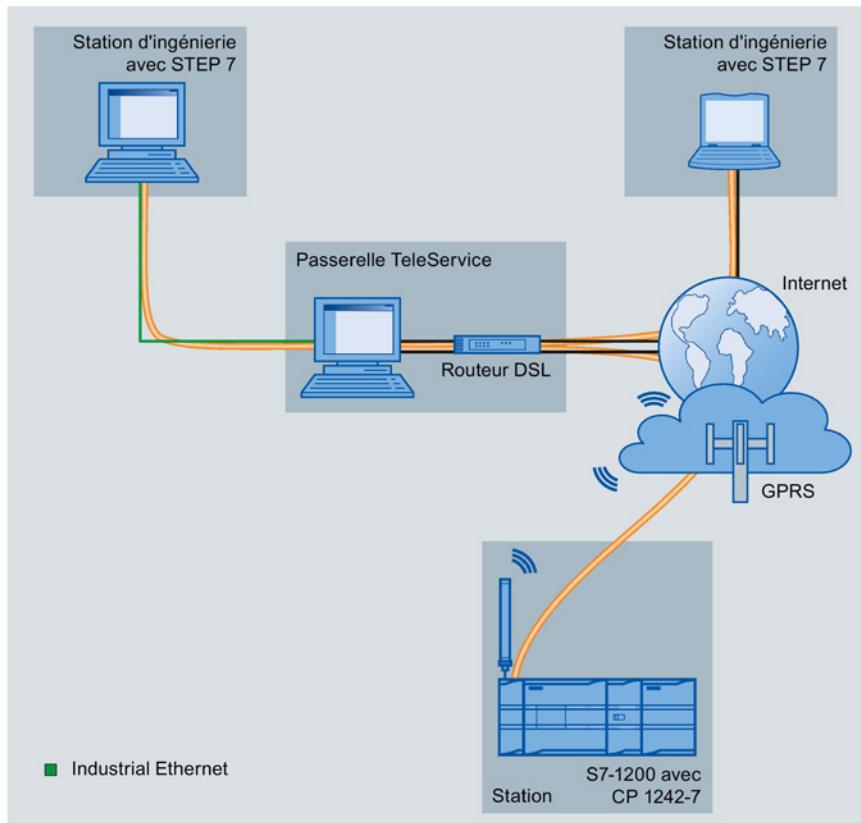


Figure 7-4 TeleService via GPRS dans une configuration avec passerelle TeleService

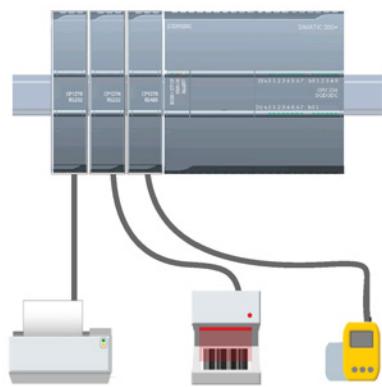
7.10 Protocoles de communication PtP, USS et Modbus

7.10.1 Communication point à point

La CPU prend en charge la communication point à point (PtP) suivante pour les protocoles série en mode caractères :

- PtP (Page 194)
- USS (Page 195)
- Modbus (Page 197)

La communication point à point offre une liberté et une souplesse maximales, mais requiert une implémentation importante dans le programme utilisateur.



La communication point à point offre de nombreuses possibilités :

- Possibilité d'envoyer des informations directement à un périphérique externe tel qu'une imprimante
- Possibilité de recevoir des informations d'autres appareils, tels que lecteurs de codes à barres, lecteurs RFID, systèmes de caméra ou de vision tiers et nombreux autres types d'appareils
- Possibilité d'échanger des informations, avec émission et réception de données, avec d'autres appareils tels que des appareils GPS, des systèmes de caméra ou de vision tiers, des modems radio et bien d'autres

Ce type de communication point à point est une communication série qui utilise des UART standard pour prendre en charge des débits et parités variés. Les modules de communication RS232 et RS422/485 (CM 1241) et le Communication Board RS485 (CB 1241) fournissent les interfaces électriques pour l'exécution de la communication point à point.

PtP sur PROFIBUS ou PROFINET

La version V4.1 de la CPU S7-1200 avec STEP 7 V13 SP1 augmente la capacité de PtP à utiliser un châssis de périphérie décentralisée PROFINET ou PROFIBUS pour communiquer avec divers appareils (lecteurs RFID, dispositif GPS et autres) :

- PROFINET (Page 156) : Vous connectez l'interface Ethernet de la CPU S7-1200 à un coupleur PROFINET. Les modules de communication PtP dans le châssis avec le coupleur peuvent ensuite fournir des communications série avec les appareils PtP.
- PROFIBUS (Page 166) : Vous insérez un module de communication PROFIBUS dans le côté gauche du châssis avec la CPU S7-1200. Vous connectez le module de communication PROFIBUS à un châssis contenant un coupleur PROFIBUS. Les modules de communication PtP dans le châssis avec le coupleur peuvent ensuite fournir des communications série avec les appareils PtP.

Pour cette raison, le S7-1200 accepte deux jeux d'instructions PtP :

- Instructions d'héritage point à point : Ces instructions existaient avec la version V4.0 du S7-1200 et fonctionnent uniquement avec des communications série à l'aide d'un module de communication CM 1241 ou d'un communication board CB 1241.
- Instructions point à point (Page 194) : Ces instructions offrent toutes les fonctionnalités des instructions d'héritage, plus la capacité à se connecter à la périphérie décentralisée PROFINET et PROFIBUS. Les instructions point à point vous permettent de configurer les communications entre les modules de communication PtP dans le châssis de périphérie décentralisée et les appareils PtP.

Remarque

Avec la version V4.1 du S7-1200, vous pouvez utiliser les instructions point à point pour tous les types de communication point à point : série, série sur PROFINET, et série sur PROFIBUS. STEP 7 fournit les instructions d'héritage point à point uniquement pour prendre en charge les programmes existants. Cependant, les instructions d'héritage fonctionnent toujours avec les CPU V4.1 ainsi qu'avec les CPU V4.0 et antérieures. Vous n'avez pas besoin de convertir les programmes antérieurs d'un jeu d'instructions à l'autre.

7.10.2 Utilisation des interfaces de communication série

Deux modules de communication (CM) et un Communication Board (CB) fournissent l'interface pour les communications PtP :

- CM 1241 RS232 (Page 445)
- CM 1241 RS422/485 (Page 444)
- CB 1241 RS485 (Page 442)

Vous pouvez connecter jusqu'à trois CM (de tout type) plus une CB pour un total de quatre interfaces de communication. Installez le CM à gauche de la CPU ou d'un autre CM.

Installez la CB sur le devant de la CPU. Pour obtenir des informations sur l'installation et la désinstallation de modules, veuillez vous référer aux instructions d'installation (Page 19).

Les interfaces de communication série ont les caractéristiques suivantes :

- Elles disposent d'un port isolé
- Elles prennent en charge les protocoles point à point
- Elles sont configurées et programmées au moyen des instructions de communication point à point
- Affichent l'activité de transmission et réception au moyen de DEL
- Affiche une DEL de diagnostic (CM uniquement)
- Sont alimentés par la CPU : connexion à une alimentation externe inutile

Reportez-vous aux caractéristiques techniques pour les interfaces de communication (Page 433).

DEL de signalisation

Les modules de communication comportent trois DEL de signalisation :

- DEL de diagnostic (DIAG) : Cette DEL clignote en rouge jusqu'à ce que la CPU y accède. Une fois la CPU sous tension, elle recherche les CM et y accède. La DEL de diagnostic commence à clignoter en vert. Cela signifie que la CPU a accédé au CM, mais ne lui a pas encore fourni la configuration. La CPU charge la configuration dans les CM configurés lorsque le programme est chargé dans la CPU. Après un chargement dans la CPU, la DEL de diagnostic du module de communication devrait émettre une lumière verte continue.
- DEL de transmission (Tx) : La DEL de transmission s'allume lorsque des données sont émises depuis le port de communication.
- DEL de réception (Rx) : Cette DEL s'allume lorsque le port de communication reçoit des données.

Le Communication Board présente des DEL de transmission (Tx) et de réception (Rx). Il n'y a pas de DEL de diagnostic.

7.10.3 Instructions point à point

Les opérations Port_Config, Send_Config et Receive_Config vous permettent de modifier la configuration à partir de votre programme utilisateur.

- Port_Config modifie les paramètres du port comme le débit en bauds.
- Send_Config modifie la configuration des paramètres de transmission en série.
- Receive_Config modifie la configuration des paramètres de réception en série dans un port de communication. Cette instruction configure les conditions qui signalent le début et la fin d'un message reçu. Les messages qui répondent à ces conditions seront reçus par l'opération Receive_P2P.

Les modifications de configuration dynamiques ne sont pas stockées de façon permanente dans la CPU. Après une mise hors puis sous tension, la configuration statique initiale de l'appareil est utilisée.

Les opérations Send_P2P, Receive_P2P et Receive_Reset contrôlent la communication PtP :

- Send_P2P transfère la mémoire tampon spécifiée au CM ou à la CB. La CPU continue à exécuter le programme utilisateur pendant que le module émet les données à la vitesse de transmission indiquée.
- Receive_P2P vérifie la présence de messages reçus dans le CM ou la CB. Si un message est disponible, il sera transféré à la CPU.
- Receive_Reset réinitialise la mémoire tampon de réception.

La mémoire tampon de chaque CM ou CB peut comporter jusqu'à 1 Koctet maximum. Cette mémoire tampon peut être utilisée pour plusieurs messages reçus.

Les instructions Signal_Set et Signal_Get sont valides uniquement pour le CM RS232. Utilisez ces instructions pour lire et régler les signaux de communication RS232.

Les instructions Get_Features et Set_Features permettent au programme de lire et de régler les fonctions du module.

7.10.4 Opérations USS

Le S7-1200 prend en charge le protocole USS et fournit des opérations qui sont spécifiquement conçues pour communiquer avec des lecteurs sur le port RS485 d'un CM ou d'une CB. Les opérations USS vous permettent de piloter l'entraînement physique et de lire ou d'écrire les paramètres de l'entraînement. Chaque CM ou CB RS485 prend en charge un maximum de 16 lecteurs.

- L'opération USS_Port_Scan gère la communication en cours entre la CPU et tous les lecteurs liés à un CM ou une CB. Insérez une opération USS_Port_Scan différente pour chaque CM ou CV dans votre application. Assurez-vous que le programme utilisateur exécute l'opération USS_Port_Scan assez rapidement pour empêcher un délai d'attente de la communication sur l'entraînement. Utilisez l'opération USS_Port_Scan dans un cycle de programme ou tout OB d'alarme.
- L'opération USS_Drive_Control accède à un entraînement spécifique sur le réseau USS. Les paramètres d'entrée et de sortie de l'opération USS_Drive_Control sont l'état et le contrôle de l'entraînement. S'il y a 16 entraînements sur le réseau, votre programme doit avoir au moins 16 opérations USS_Drive_Control avec une opération pour chaque entraînement.

Assurez-vous que la CPU exécute l'opération USS_Drive_Control au débit requis pour contrôler les fonctions de l'entraînement. Utilisez l'opération USS_Drive_Control uniquement dans un OB de cycle de programme.

- Les opérations USS_Read_Param et USS_Write_Param lisent et écrivent les paramètres de fonctionnement de l'entraînement à distance. Ces paramètres pilotent le fonctionnement interne de l'entraînement. Vous trouverez la définition de ces paramètres dans le manuel de l'entraînement.

Votre programme peut contenir le nombre d'instructions nécessaires. Cependant, une seule requête de lecture ou d'écriture peut être active pour un pilote quelconque à un instant donné. Utilisez les opérations USS_Read_Param et USS_Write_Param uniquement dans un OB de cycle de programme.

Un DB d'instance contient un stockage temporaire et des mémoires tampons pour tous les entraînements sur le réseau USS connectés à chaque CM ou CB. Les instructions USS pour un pilote utilisent le DB d'instance pour partager les informations.

Calcul du temps requis pour communiquer avec l'entraînement

Les communications avec le pilote sont asynchrone par rapport au cycle de la CPU. Typiquement, la CPU exécute plusieurs cycles avant qu'une transaction de communication avec un pilote ne soit terminée.

L'intervalle USS_Port_Scan est le temps requis pour une transaction avec un entraînement. Le tableau ci-dessous montre l'intervalle USS_Port_Scan minimum pour chaque vitesse de transmission. Appeler la fonction USS_Port_Scan plus fréquemment que l'intervalle USS_Port_Scan n'augmente pas le nombre de transactions. Le délai d'attente de l'entraînement est la quantité de temps qui pourrait être utilisée pour une transaction si des erreurs de communication entraînaient 3 tentatives pour achever la transaction. Par défaut, la bibliothèque du protocole USS exécute automatiquement jusqu'à 2 nouvelles tentatives pour chaque transaction.

Tableau 7- 14 Calcul des exigences en temps

Débit en bauds	Intervalle d'appel USS_Port_Scan minimum calculé (millisecondes)	Délai d'attente de message par entraînement (millisecondes)
1200	790	2370
2400	405	1215
4800	212.5	638
9600	116.3	349
19200	68.2	205
38400	44.1	133
57600	36.1	109
115200	28.1	85

7.10.5 Opérations Modbus

La CPU prend en charge la communication Modbus par le biais de différents réseaux :

- Modbus RTU (Remote Terminal Unit : unité terminale distante) est un protocole de communication de réseau standard qui utilise la liaison électrique RS232 ou RS485 pour le transfert de données série entre dispositifs réseau Modbus. Vous pouvez ajouter des ports de réseau point à point (PtP) à une CPU avec un CM RS232 ou RS485 ou un CB RS485.

Modbus RTU utilise un réseau maître/esclave où toutes les communications sont déclenchées par un maître unique et où les esclaves peuvent uniquement répondre à une demande du maître. Le maître envoie une demande à une adresse d'esclave et seule cette adresse d'esclave répond à la commande.

- Modbus TCP (Transmission Control Protocol : protocole de gestion de transmission) est un protocole de communication de réseau standard qui utilise le connecteur PROFINET sur la CPU pour la communication TCP/IP. Aucun module matériel de communication supplémentaire n'est nécessaire.

Modbus TCP utilise des liaisons client-serveur comme voie de communication Modbus. Plusieurs liaisons client-serveur peuvent exister en plus de la liaison entre STEP 7 et la CPU. Les liaisons mixtes client et serveur sont prises en charge jusqu'au nombre maximum de liaisons autorisé par le modèle de CPU. Chaque liaison MB_SERVER doit utiliser un DB d'instance et un numéro de port IP uniques. Une seule liaison est acceptée par port IP. Chaque instruction MB_SERVER (avec son DB d'instance et son port IP uniques) doit être exécutée individuellement pour chaque liaison.

ATTENTION

Si un attaquant peut accéder physiquement à vos réseaux, il sera éventuellement en mesure de lire et écrire des données.

TIA Portal, la CPU et les IHM (excepté les IHM utilisant GET/PUT) utilisent une communication sécurisée qui protège contre les attaques de type "Replay" et "homme du milieu". Une fois la communication établie, l'échange de messages signés a lieu en texte clair, ce qui permet à un attaquant de lire les données, mais l'échange est protégé contre l'écriture non autorisée de données. C'est TIA Portal, et non le processus de communication, qui crypte les données des blocs avec protection Know-How.

Toutes les autres formes de communication (échange de données E/S via PROFIBUS, PROFINET, AS-i ou tout autre bus E/S, GET/PUT, bloc T et modules de communication (CM)) ne sont pas dotées de fonctionnalités de sécurité. Vous devez protéger ces formes de communication en limitant l'accès physique. Si un attaquant peut accéder physiquement à vos réseaux en utilisant ces formes de communication, il sera en mesure de lire et écrire des données.

Pour obtenir des informations et des recommandations de sécurité, veuillez vous reporter à nos "Directives d'exploitation pour la sécurité industrielle" sur le site Service et Assistance : www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf (http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf)

Remarque

Modbus TCP ne fonctionne correctement qu'avec une version de firmware CPU V1.02 ou plus. Tenter d'exécuter les instructions Modbus avec une version de firmware antérieure provoque une erreur.

Tableau 7- 15 Instructions Modbus

Type de communication	Instruction
Modbus RTU (RS232 ou RS485)	Modbus_Comm_Load : une exécution de Modbus_Comm_Load sert à configurer les paramètres de port point à point, tels que la vitesse de transmission, la parité et le contrôle de flux. Une fois que le port CPU est configuré pour le protocole Modbus RTU, il peut être utilisé uniquement par les instructions Modbus_Master ou Modbus_Slave.
	Modbus_Master : l'instruction Modbus maître permet à la CPU d'agir en tant que maître Modbus RTU et de communiquer avec un ou plusieurs esclaves Modbus.
	Modbus_Slave : l'instruction Modbus esclave permet à la CPU d'agir en tant qu'esclave Modbus RTU et de communiquer avec un maître Modbus.
Modbus TCP (PROFINET)	MB_CLIENT : Etablir une liaison TCP client-serveur, envoyer un message de commande, recevoir la réponse et gérer la déconnexion du serveur
	MB_SERVER : Se connecter à un client Modbus TCP sur demande, recevoir un message Modbus et envoyer la réponse

Les instructions Modbus n'utilisent pas d'événements d'alarme de communication pour piloter le processus de communication. Votre programme doit interroger les conditions émission et réception achevées des instructions Modbus_Master / Modbus_Slave ou MB_CLIENT / MB_SERVER.

Un client Modbus TCP (maître) doit gérer la liaison client-serveur à l'aide du paramètre DISCONNECT. Les actions de base Modbus client sont présentées ci-dessous.

1. Etablir une liaison vers l'adresse IP et le numéro de port IP d'un serveur particulier (esclave)
2. Déclencher l'émission client de messages Modbus et recevoir les réponses du serveur
3. Si désiré, déclencher la déconnexion du client et du serveur pour permettre la liaison avec un serveur différent

Le PID, c'est simple

STEP 7 fournit les instructions PID suivantes pour la CPU S7-1200 :

- L'instruction PID_Compact sert à la régulation de processus industriels à grandeurs d'entrée et de sortie continues.
- L'instruction PID_3Step sert à la régulation de dispositifs actionnés par moteur, tels que des vannes, qui nécessitent des signaux discrets pour la commande d'ouverture et de fermeture.
- L'instruction PID_Temp offre un régulateur PID universel qui permet de gérer des exigences spécifiques en matière de régulation de la température.

Remarque

Les modifications que vous apportez à la configuration PID et que vous chargez à l'état MARCHE ne prennent effet qu'après un passage de la CPU de l'état ARRET à l'état MARCHE. Les modifications que vous apportez dans la boîte de dialogue des "paramètres PID" à l'aide de la "Commande de la valeur de début" prennent effet immédiatement.

Les trois instructions PID (PID_Compact, PID_3Step, et PID_Temp) peuvent calculer les actions P, I et D lors de la mise en route (si "l'optimisation préalable" est configurée). Vous pouvez également configurer l'instruction pour une "optimisation fine" qui vous permet d'optimiser les paramètres. Vous n'avez pas besoin de déterminer les paramètres manuellement.

Remarque

Exécutez l'instruction PID à intervalles constants de la période d'échantillonnage (de préférence dans un OB d'alarme cyclique).

Comme la boucle PID a besoin d'un certain temps pour réagir aux variations de la valeur de réglage, ne calculez pas la valeur de réglage à chaque cycle. N'exédez pas l'instruction PID dans l'OB de cycle de programme principal (tel que l'OB 1).

La période d'échantillonnage de l'algorithme PID représente le temps entre deux calculs de la valeur de réglage. La valeur de réglage est calculée pendant l'autoréglage et arrondie à un multiple de la période. Toutes les autres fonctions de l'instruction PID sont exécutées à chaque appel.

Algorithme PID

Le régulateur PID (actions proportionnelle/intégrale/dérivée) mesure l'intervalle de temps entre deux appels, puis évalue les résultats pour surveiller la période d'échantillonnage. Une valeur moyenne de la période d'échantillonnage est générée à chaque changement de mode ainsi qu'à la mise en route initiale. Cette valeur sert de référence à la fonction de surveillance et est utilisée pour les calculs. La surveillance porte sur le temps de mesure actuel entre deux appels et sur la valeur moyenne de la période d'échantillonnage définie du régulateur.

La valeur de réglage pour le régulateur PID est constituée de trois actions :

- Action proportionnelle P : La valeur de réglage calculée par l'action P est proportionnelle à l'écart entre la consigne et la mesure.
- Action intégrale I : La valeur de réglage calculée par l'action I augmente proportionnellement à la durée de l'écart entre la consigne et la mesure pour finalement compenser l'écart.
- Action dérivée D : La valeur de réglage calculée par l'action D augmente avec l'accroissement de la vitesse de modification de l'écart entre la consigne et la mesure. La valeur de réglage est synchronisée sur la consigne le plus rapidement possible.

Le régulateur PID utilise la formule suivante pour calculer la valeur de réglage pour l'instruction PID_Compact.

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

y	Valeur de réglage	x	Mesure
w	Consigne	s	Opérateur de Laplace
K _p	Gain proportionnel (action P)	a	Coefficient de retard de l'action dérivée (action D)
T _I	Temps d'intégration (action I)	b	Pondération de l'action proportionnelle (action P)
T _D	Temps de dérivation (action D)	c	Pondération de l'action dérivée (action D)

Le régulateur PID utilise la formule suivante pour calculer la valeur de réglage pour l'instruction PID_3Step.

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

y	Valeur de réglage	x	Mesure
w	Consigne	s	Opérateur de Laplace
K _p	Gain proportionnel (action P)	a	Coefficient de retard de l'action dérivée (action D)
T _I	Temps d'intégration (action I)	b	Pondération de l'action proportionnelle (action P)
T _D	Temps de dérivation (action D)	c	Pondération de l'action dérivée (action D)

8.1

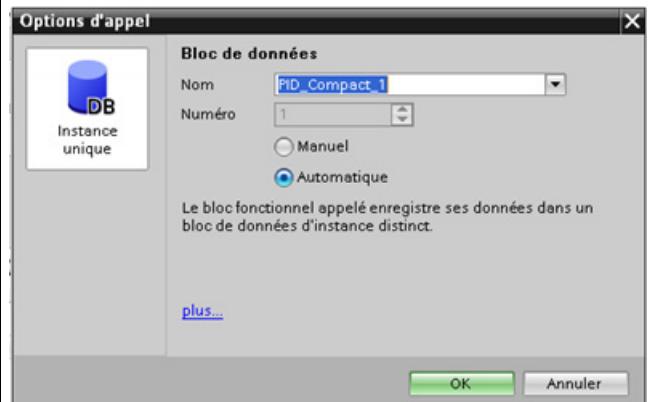
Insertion de l'instruction PID et de l'objet technologique

STEP 7 fournit deux instructions pour la régulation PID :

- L'instruction PID_Compact et son objet technologique associé fournissent un régulateur PID universel optimisable. L'objet technologique contient tous les paramètres pour la boucle de régulation.
- L'instruction PID_3Step et son objet technologique associé fournissent un régulateur PID avec des paramètres spécifiques pour les vannes commandées par moteur. L'objet technologique contient tous les paramètres pour la boucle de régulation. Le régulateur PID_3Step comporte deux sorties booléennes supplémentaires.

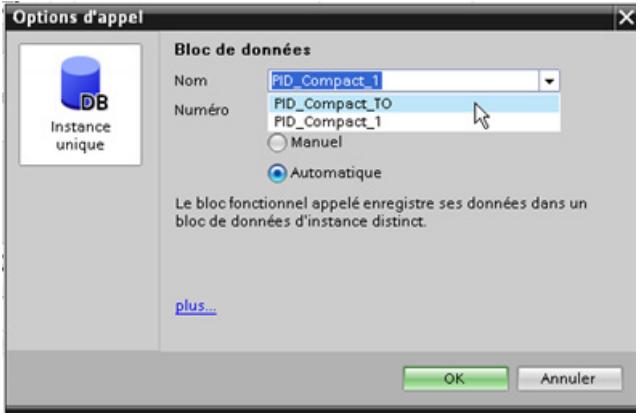
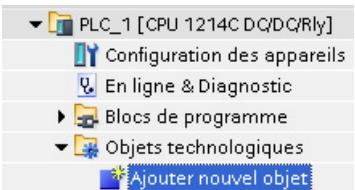
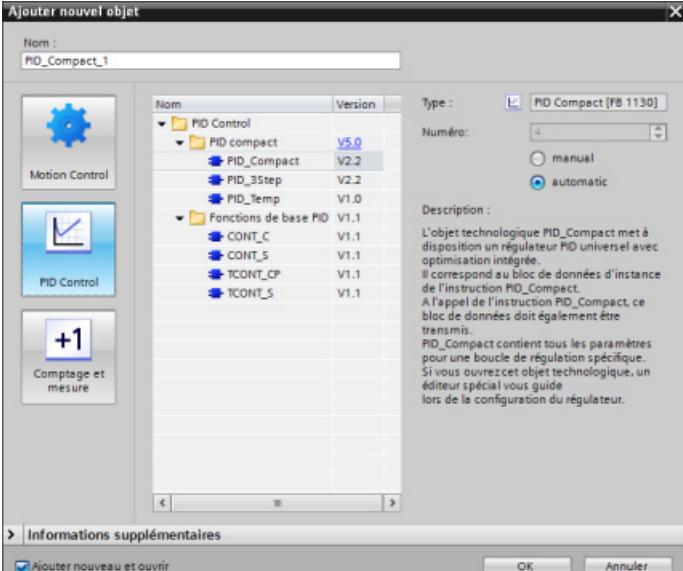
Une fois l'objet technologique créé, vous devez configurer les paramètres (Page 231). Vous ajustez également les paramètres d'autooptimisation ("optimisation préalable" à la mise en route ou "optimisation fine" manuelle) pour mettre en service le régulateur PID (Page 249).

Tableau 8- 1 Insertion de l'instruction PID et de l'objet technologique

<p>Lorsque vous insérez une instruction PID dans votre programme utilisateur, STEP 7 crée automatiquement un objet technologique et un DB d'instance pour l'instruction. Le DB d'instance contient tous les paramètres utilisés par l'instruction PID. Chaque instruction PID doit avoir son propre DB d'instance unique pour fonctionner correctement.</p> <p>Après l'insertion de l'instruction PID et la création de l'objet technologique et du DB d'instance, vous configurez les paramètres pour l'objet technologique (Page 231).</p>	
--	---

8.1 Insertion de l'instruction PID et de l'objet technologique

Tableau 8- 2 (Facultatif) Création d'un objet technologique dans le navigateur du projet

<p>Vous pouvez également créer des objets technologiques pour votre projet avant d'insérer l'instruction PID. Si vous avez créé l'objet technologique avant d'insérer une instruction PID dans votre programme utilisateur, vous pourrez sélectionner cet objet technologique à l'insertion de l'instruction PID.</p>	
<p>Pour créer un objet technologique, double-cliquez sur l'icône "Ajouter nouvel objet" dans le navigateur du projet.</p>	
<p>Cliquez sur l'icône "Régulation" et sélectionnez l'objet technologique correspondant au type de régulateur PID (PID_Compact ou PID_3Step). Vous pouvez créer un nom facultatif pour l'objet technologique.</p> <p>Cliquez sur "OK" pour créer l'objet technologique.</p>	

8.2 Instruction PID_Compact

L'instruction PID_Compact fournit un régulateur PID universel avec fonction d'autooptimisation intégrée pour les modes automatique et manuel.

Tableau 8- 3 Instruction PID_Compact

CONT/LOG	SCL	Description
<p>%DB2 "PID_Compact_1" PID_Compact</p> <p>EN ScaledInput Output Output_PER Output_PWM SetpointLimit_H SetpointLimit_L InputWarning_H InputWarning_L State Error ErrorBits</p>	<pre>"PID_Compact_1"(Setpoint:= _real_in_, Input:= _real_in_, Input_PER:= _word_in_, Disturbance:= _real_in_, ManualEnable:= _bool_in_, ManualValue:= _real_in_, ErrorAck:= _bool_in_, Reset:= _bool_in_, ModeActivate:= _bool_in_, Mode:= _int_in_, ScaledInput=> _real_out_, Output=> _real_out_, Output_PER=> _word_out_, Output_PWM=> _bool_out_, SetpointLimit_H=> _bool_out_, SetpointLimit_L=> _bool_out_, InputWarning_H=> _bool_out_, InputWarning_L=> _bool_out_, State=> _int_out_, Error=> _bool_out_, ErrorBits=> dword_out);</pre>	<p>PID_Compact fournit un régulateur PID à fonction d'autooptimisation pour les modes automatique et manuel. PID_Compact est un régulateur PID T1 avec anti-emballement (anti-windup) et pondération des actions P et D.</p>

- 1 STEP 7 crée automatiquement l'objet technologique et le DB d'instance lorsque vous insérez l'instruction. Le DB d'instance contient les paramètres de l'objet technologique.
- 2 Dans l'exemple SCL, "PID_Compact_1" est le nom du DB d'instance.

Tableau 8- 4 Types de données pour les paramètres

Paramètre et type	Type de données	Description
Setpoint	IN	Real Consigne du régulateur PID en mode automatique. (valeur par défaut : 0,0)
Input	IN	Real Une variable du programme utilisateur est utilisée comme source pour la mesure. (valeur par défaut : 0,0) Si vous utilisez le paramètre Input alors il faut définir Config.InputPerOn = FALSE.
Input_PER	IN	Word Une entrée analogique est utilisée comme source pour la mesure. (valeur par défaut : W#16#0) Si vous utilisez le paramètre Input_PER alors il faut définir Config.InputPerOn = TRUE.
Disturbance	IN	Real Variable de perturbation ou valeur de pré-régulation
ManualEnable	IN	Bool Active ou désactive le mode de fonctionnement manuel. (valeur par défaut : FALSE) : <ul style="list-style-type: none"> • Un front de FALSE à TRUE active le mode manuel", tant que Etat = 4, le mode reste inchangé. <p>Tant que ManualEnable est VRAI, vous ne pouvez pas modifier l'état de fonctionnement à l'aide d'un front montant pour ModeActivate ou utiliser la boîte de dialogue de mise en service.</p> <ul style="list-style-type: none"> • Un front TRUE à FALSE active l'état de fonctionnement qui est affecté par Mode. <p>Note : Nous vous recommandons de modifier l'état de fonctionnement à l'aide de ModeActivate uniquement.</p>
ManualValue	IN	Real Valeur de sortie pour le mode manuel. (valeur par défaut : 0,0) Vous pouvez utiliser des valeurs à partir Config.OutputLowerLimit à Config.OutputUpperLimit.
ErrorAck	IN	Bool Réinitialise le ErrorBits et les messages d'alerte. Front de FALSE à TRUE
Reset	IN	Bool Redémarre le régulateur. (valeur par défaut : FALSE) : <ul style="list-style-type: none"> • Front de FALSE à TRUE : <ul style="list-style-type: none"> – Passe en mode "inactif" – Réinitialise les ErrorBits et les sorties d'alerte – Efface l'action intégrale – Maintient les paramètres PID • Tant que Reset à la valeur TRUE, PID_Compact reste en mode "Inactif" (Etat = 0). • Front de TRUE à FALSE : <ul style="list-style-type: none"> – PID_Compact passe dans l'état de fonctionnement qui est enregistré dans le paramètre Mode.
ModeActivate	IN	Bool Le PID_Compact passe dans l'état de fonctionnement qui est enregistré dans le paramètreMode. Front de FALSE à TRUE :
Mode	IN	Int Le mode PID souhaité ; Activé sur le front montant de l'entrée Mode Activate .
ScaledInput	OUT	Real Mesure mise à l'échelle. (valeur par défaut : 0,0)

Paramètre et type		Type de données	Description
Output ¹	OUT	Real	Valeur de sortie pour le mode REEL. (valeur par défaut : 0,0)
Output_PER ¹	OUT	Word	Valeur de réglage analogique. (valeur par défaut : W#16#0)
Output_PWM ¹	OUT	Bool	Valeur de réglage pour la modulation de largeur d'impulsion. (valeur par défaut : FALSE) Les durées Marche et Arrêt forment la valeur de réglage.
SetpointLimit_H	OUT	Bool	Limite supérieure de la consigne. (valeur par défaut : FALSE) Si SetpointLimit_H a la valeur TRUE, la limite supérieure absolue de la consigne est atteinte.(Setpoint ≥ Config.SetpointUpperLimit). La consigne est limitée à Config.SetpointUpperLimit.
SetpointLimit_L	OUT	Bool	Limite inférieure de la consigne. (valeur par défaut : FALSE) Si SetpointLimit_L a la valeur TRUE, la limite inférieure absolue de la consigne est atteinte.(Setpoint ≤ Config.SetpointLowerLimit). La consigne est limitée à Config.SetpointLowerLimit.
InputWarning_H	OUT	Bool	Si InputWarning_H a la valeur TRUE, la mesure a atteint ou dépassé la limite d'alerte supérieure. (valeur par défaut : FALSE)
InputWarning_L	OUT	Bool	Si InputWarning_L a la valeur TRUE, la mesure de processus a atteint ou est tombée en dessous de la limite d'alerte inférieure. (valeur par défaut : FALSE)
State	OUT	Int	Mode de fonctionnement en cours du régulateur PID. (valeur par défaut : 0) Vous pouvez modifier l'état de fonctionnement à l'aide du paramètre d'entrée Mode et un front montant pour ModeActivate : <ul style="list-style-type: none">• State = 0 : Inactif• State = 1 : optimisation préalable• State = 2 : Optimisation fine manuelle• State = 3 : Mode automatique• State = 4 : Mode manuel• State = 5 : Valeur de réglage de remplacement avec surveillance d'erreur
Error	OUT	Bool	Si Error a la valeur TRUE, au moins un message d'erreur est en attente dans ce cycle. (valeur par défaut : FALSE) Note : Le paramètre Error dans V1.x PID était le champ ErrorBits qui contenait les codes d'erreurs. C'est maintenant une signalisation booléenne qui indique qu'une erreur est survenue.
ErrorBits	OUT	DWord	Le tableau des PID_Compact paramètres de l'instruction ErrorBits présente les messages d'erreur (Page 207) qui sont en attente. (valeur par défaut : DW#16#0000 (aucune erreur)). ErrorBits est rémanent et réinitialisé sur un front montant pour Reset ou ErrorAck. Note : Dans V1.x, le paramètre ErrorBits a été défini comme le paramètre Error et n'existe pas.

¹ Vous pouvez utiliser les sorties de la Output, Output_PER et Output_PWM des paramètres en parallèle.

Fonctionnement du régulateur PID_Compact

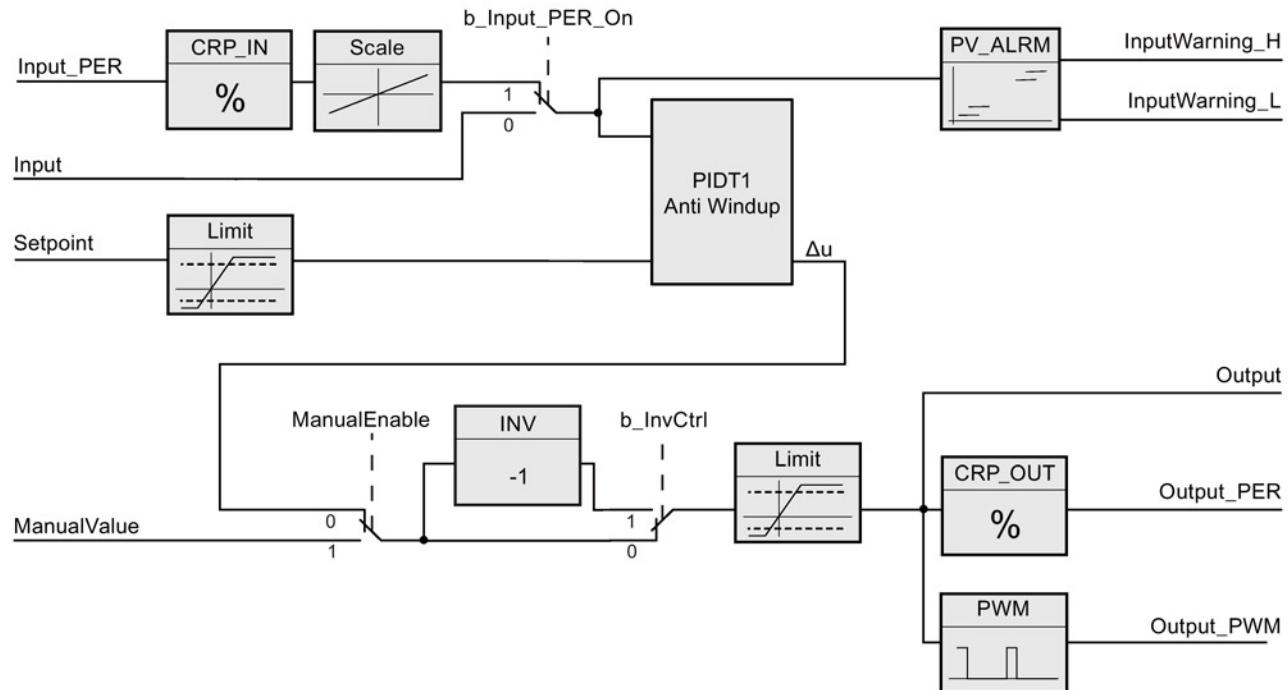


Figure 8-1 Fonctionnement du régulateur PID_Compact

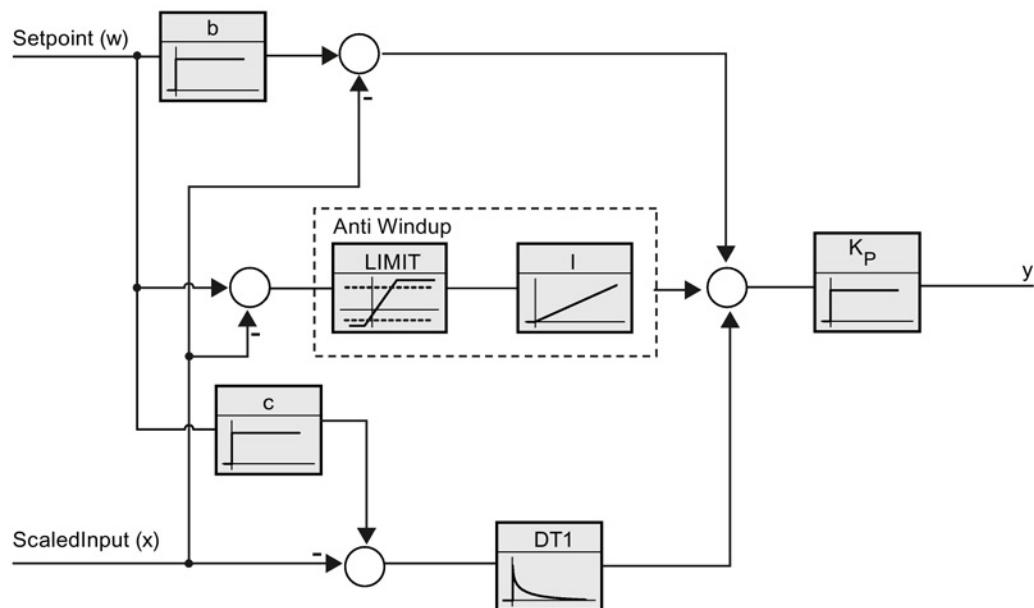


Figure 8-2 Fonctionnement du régulateur PID_Compact en tant que régulateur PIDT1 avec anti-emballlement

8.3 Paramètre ErrorBits de l'instruction PID_Compact

Si plusieurs erreurs sont en attente, les valeurs des codes d'erreur sont données au moyen d'une addition binaire. Ainsi, l'affichage du code d'erreur 0003 signifie que les erreurs 0001 et 0002 sont également en attente.

Tableau 8- 5 Paramètre ErrorBits de l'instruction PID_Compact

ErrorBit (DW#16#...)	Description
0000	Pas d'erreur
0001 ^{1, 2}	Le paramètre Input se situe en dehors des limites de la mesure. Input > Config.InputUpperLimit Input < Config.InputLowerLimit
0002 ^{2, 3}	Valeur invalide pour le paramètre Input_PER. Vérifiez s'il y a une erreur en attente au niveau de l'entrée analogique.
0004 ⁴	Erreur pendant l'optimisation fine. L'oscillation de la mesure n'a pas pu être conservée.
0008 ⁴	Erreur au démarrage de l'optimisation préalable. La mesure est trop proche de la consigne. Démarrer l'optimisation fine.
0010 ⁴	La consigne a été modifiée pendant l'optimisation. Note : Vous pouvez définir la fluctuation permise sur la consigne à la variable CancelTuningLevel.
0020	L'optimisation préalable n'est pas permise pendant l'optimisation fine. Note : Si ActivateRecoverMode à la valeur TRUE avant que l'erreur ne survienne, PID_Compact reste en mode optimisation fine.
0080 ⁴	Erreur pendant l'optimisation préalable. Configuration incorrecte des limites de la valeur de réglage. Vérifiez si les limites de la valeur de réglage sont configurées correctement et correspondent à la logique du régulateur.
0100 ⁴	Une erreur pendant l'optimisation fine a entraîné des paramètres invalides.
0200 ^{2, 3}	Valeur invalide pour le paramètre Input. La valeur a un format numérique invalide.
0400 ^{2, 3}	Le calcul de la valeur de réglage a échoué. Vérifiez les paramètres PID.
0800 ^{1, 2}	Erreur de période d'échantillonnage : PID_Compact n'est pas appelé dans la période d'échantillonnage de l'OB d'alarme cyclique.
1000 ^{2, 3}	Valeur invalide pour le paramètre Setpoint. La valeur a un format numérique invalide.
10000	Valeur invalide pour le paramètre ManualValue. La valeur a un format numérique invalide. Note : Si ActivateRecoverMode à la valeur TRUE avant que l'erreur ne survienne, PID_Compact utilise SubstituteOutput comme valeur de réglage. Dès que vous affectez une valeur invalide dans le paramètre ManualValue, PID_Compact l'utilise comme valeur de réglage.

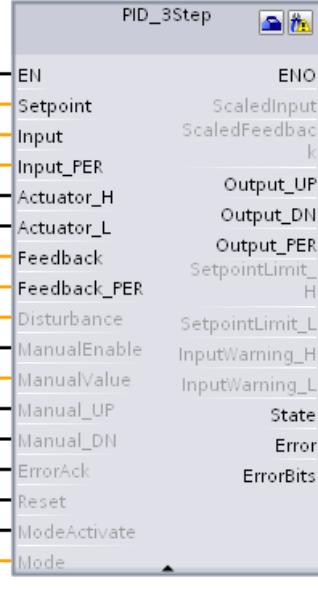
ErrorBit (DW#16#...)	Description
20000	<p>Valeur invalide pour la variable SubstituteValue : La valeur a un format numérique invalide.</p> <p>PID_Compact utilise la limite inférieure de la valeur de réglage comme valeur de réglage.</p> <p>Note : Si le mode automatique était actif avant l'apparition de l'erreur, ActivateRecoverMode a la valeur TRUE et que l'erreur n'est plus en attente, PID_Compact repasse en mode automatique.</p>
40000	<p>Valeur invalide pour le paramètre Disturbance. La valeur a un format numérique invalide.</p> <p>Note : Si le mode automatique était actif et si ActivateRecoverMode a la valeur FALSE avant l'apparition de l'erreur, Disturbance est défini sur zéro. PID_Compact reste en mode automatique.</p> <p>Note : Si l'optimisation préalable ou l'optimisation fine était active et ActivateRecoverMode a la valeur TRUE avant que l'erreur ne se produise, PID_Compact passe dans l'état de fonctionnement qui est enregistré pour le paramètre Mode. Si une perturbation dans la phase actuelle n'a pas d'effet sur la valeur de réglage, l'optimisation n'est pas annulée.</p>

- ¹ Note : Si l'état automatique était actif avant que l'erreur ne se produise et ActivateRecoverMode a la valeur TRUE, PID_Compact reste en mode automatique.
- ² Note : Si l'optimisation préalable ou l'optimisation fine était active avant que l'erreur ne se produise et si ActivateRecoverMode a la valeur TRUE, PID_Compact passe dans l'état de fonctionnement qui est enregistré pour le paramètre Mode.
- ³ Note : Si le mode automatique était actif avant que l'erreur ne se produise et si ActivateRecoverMode a la valeur TRUE, PID_Compact délivre la valeur de réglage de remplacement configurée. Dès que l'erreur n'est plus en attente, PID_Compact repasse en mode automatique.
- ⁴ Note : Si ActivateRecoverMode a la valeur TRUE avant que l'erreur ne se produise, PID_Compact annule l'optimisation et passe dans l'état de fonctionnement qui est enregistré pour le paramètre Mode.

8.4 Instruction PID_3Step

L'instruction PID_3Step configure un régulateur PID à fonction d'autooptimisation qui a été optimisé pour des vannes et actionneurs commandés par moteur.

Tableau 8- 6 Instruction PID_3Step

CONT/LOG	SCL	Description
	<pre> "PID_3Step_1"(Setpoint:= _real_in_, Input:= _real_in_, ManualValue:= _real_in_, Feedback:= _real_in_, InputPer:= _word_in_, FeedbackPer:= _word_in_, Disturbance:= _real_in_, ManualEnable:= _bool_in_, ManualUP:= _bool_in_, ManualDN:= _bool_in_, ActuatorH:= _bool_in_, ActuatorL:= _bool_in_, ErrorAck:= _bool_in_, Reset:= _bool_in_, ModeActivate:= _bool_in_, Mode:= _int_in_, ScaledInput=> _real_out_, ScaledFeedback=> _real_out_, ErrorBits=> _dword_out_, OutputPer=> _word_out_, State=> _int_out_, OutputUP=> _bool_out_, OutputDN=> _bool_out_, SetpointLimitH=> _bool_out_, SetpointLimitL=> _bool_out_, InputWarningH=> _bool_out_, InputWarningL=> _bool_out_, Error=> _bool_out_, ErrorBits=> _dword_out_); </pre>	<p>PID_3Step configure un régulateur PID à fonction d'autooptimisation qui a été optimisé pour des vannes et actionneurs commandés par moteur. Cette instruction fournit deux sorties booléennes.</p> <p>PID_3Step est un régulateur PID T1 avec anti-emballement et pondération des actions P et D.</p>

- 1 STEP 7 crée automatiquement l'objet technologique et le DB d'instance lorsque vous insérez l'instruction. Le DB d'instance contient les paramètres de l'objet technologique.
- 2 Dans l'exemple SCL, "PID_3Step_1" est le nom du DB d'instance.

Tableau 8- 7 Types de données pour les paramètres

Paramètre et type	Type de données	Description
Setpoint	IN	Real Consigne du régulateur PID en mode automatique. (valeur par défaut : 0.0)
Input	IN	Real Une variable du programme utilisateur est utilisée comme source pour la mesure. (valeur par défaut : 0.0) Si vous utilisez le paramètre Input alors il faut définir Config.InputPerOn = FALSE.
Input_PER	IN	Word Une entrée analogique est utilisée comme source pour la mesure. (valeur par défaut : W#16#0) Si vous utilisez le paramètre Input_PER alors il faut définir Config.InputPerOn = TRUE.
Actuator_H	IN	Bool Signalisation de position numérique de la vanne pour la butée supérieure Si Actuator_H = TRUE, la vanne est à la butée supérieure et n'est plus déplacée dans ce sens. (valeur par défaut : FALSE)
Actuator_L	IN	Bool Signalisation de position numérique de la vanne pour la butée inférieure Si Actuator_L = TRUE, la vanne est à la butée inférieure et n'est plus déplacée dans ce sens. (valeur par défaut : FALSE)
Feedback	IN	Real Signalisation de position de la vanne. (valeur par défaut : 0.0) Si vous utilisez le paramètre Feedback alors il faut définir Config.FeedbackPerOn = FALSE.
Feedback_PER	IN	Int Signalisation analogique de la position de la vanne. (valeur par défaut : W#16#0) Si vous utilisez le paramètre Feedback_PER, vous devez définir Config.FeedbackPerOn = TRUE. Feedback_PER est mise à l'échelle, en fonction des variables suivantes : <ul style="list-style-type: none">Config.FeedbackScaling.LowerPointInConfig.FeedbackScaling.UpperPointInConfig.FeedbackScaling.LowerPointOutConfig.FeedbackScaling.UpperPointOut
Disturbance	IN	Real Variable de perturbation ou valeur de pré-régulation
ManualEnable	IN	Bool Active ou désactive le mode de fonctionnement manuel. (valeur par défaut : FALSE): <ul style="list-style-type: none">Un front de FALSE à TRUE active le mode manuel", tant que Etat = 4, le mode reste inchangé. Tant que ManualEnable est VRAI, vous ne pouvez pas modifier l'état de fonctionnement à l'aide d'un front montant pour ModeActivate ou utiliser la boîte de dialogue de mise en service. <ul style="list-style-type: none">Un front TRUE à FALSE active l'état de fonctionnement qui est affecté par Mode. Note : Nous vous recommandons de modifier l'état de fonctionnement à l'aide de ModeActivate uniquement.

Paramètre et type		Type de données	Description
ManualValue	IN	Real	<p>Valeur de réglage pour le mode manuel. (valeur par défaut : 0.0)</p> <p>En mode manuel, vous indiquez la position absolue de la vanne. ManualValue est évalué uniquement si vous utilisez OutputPer, ou si la signalisation de position est disponible :</p>
ManualUP	IN	Bool	<ul style="list-style-type: none"> • Manual_UP = TRUE: <ul style="list-style-type: none"> – La vanne est ouverte même si vous utilisez Output_PER ou une signalisation de position. La vanne n'est plus déplacée si la butée supérieure a été atteinte. – Voir aussi Config.VirtualActuatorLimit • Manual_UP = FALSE: <ul style="list-style-type: none"> – Si vous utilisez Output_PER ou une signalisation de position, la vanne est déplacée sur ManualValue. Sinon, la vanne n'est plus déplacée. <p>Note : Si Manual_UP et Manual_DN ne sont pas paramétrés sur TRUE simultanément, la vanne n'est pas déplacée.</p>
ManualDN	IN	Bool	<ul style="list-style-type: none"> • Manual_DN = TRUE: <ul style="list-style-type: none"> – La vanne est ouverte même si vous utilisez Output_PER ou une signalisation de position. La vanne n'est plus déplacée si la butée supérieure a été atteinte. – Voir aussi Config.VirtualActuatorLimit • Manual_DN = FALSE: <ul style="list-style-type: none"> – Si vous utilisez Output_PER ou une signalisation de position, la vanne est déplacée sur ManualValue. Sinon, la vanne n'est plus déplacée.
ErrorAck	IN	Bool	Réinitialise le ErrorBits et les messages d'alerte. Front de FALSE à TRUE
Reset	IN	Bool	<p>Redémarre le régulateur. (valeur par défaut : FALSE):</p> <ul style="list-style-type: none"> • Front de FALSE à TRUE : <ul style="list-style-type: none"> – Passe en mode "inactif" – Réinitialise les ErrorBits et les sorties d'alerte – Efface l'action intégrale – Maintient les paramètres PID • Tant que Reset à la valeur TRUE, PID_3Step reste en mode "Inactif" (Etat = 0). • Front de TRUE à FALSE : <ul style="list-style-type: none"> – PID_3Step passe dans l'état de fonctionnement qui est enregistré dans le paramètre Mode.
ModeActivate	IN	Bool	Le PID_3Step passe dans l'état qui est enregistré pour le paramètre Mode. Front de FALSE à TRUE :
Mode	IN	Int	Le mode PID souhaité ; Activé sur le front montant de l'entrée Mode Activate .
ScaledInput	OUT	Real	Mesure mise à l'échelle

Paramètre et type	Type de données	Description
ScaledFeedback	OUT	Real Signalisation de position de vanne mise à l'échelle Note : Pour un actionneur sans signalisation de position, la position de l'actionneur indiquée par ScaledFeedback est très imprécise. ScaledFeedback peut seulement être utilisée que pour une estimation approximative de la position actuelle dans ce cas.
Output_UP	OUT	Bool Valeur de réglage TOR pour l'ouverture de la vanne. (valeur par défaut : FALSE) Si Config.OutputPerOn a la valeur FALSE, le paramètre Output_UP est utilisé.
Output_DN	OUT	Bool Valeur de réglage TOR pour la fermeture de la vanne. (valeur par défaut : FALSE) Si Config.OutputPerOn a la valeur FALSE, le paramètre Output_DN est utilisé.
Output_PER	OUT	Word Valeur de réglage analogique. Si Config.OutputPerOn a la valeur TRUE, le paramètre Output_PER est utilisé.
SetpointLimitH	OUT	Bool Limite supérieure de la consigne. (valeur par défaut : FALSE) Si SetpointLimitH a la valeur TRUE, la limite supérieure absolue de la consigne est atteinte (Setpoint \geq Config.SetpointUpperLimit). Note : La consigne est limitée à (Setpoint \geq Config.SetpointUpperLimit).
SetpointLimitL	OUT	Bool Limite inférieure de la consigne. (valeur par défaut : FALSE) Si SetpointLimitL a la valeur TRUE, la limite inférieure absolue de la consigne est atteinte (Setpoint \geq Config.SetpointLowerLimit). Note : La consigne est limitée à (Setpoint \geq Config.SetpointLowerLimit).
InputWarningH	OUT	Bool Si InputWarningH a la valeur TRUE, la grandeur d'entrée a atteint ou dépassé la limite d'alerte supérieure. (valeur par défaut : FALSE)
InputWarningL	OUT	Bool Si InputWarningL a la valeur TRUE, la grandeur d'entrée a atteint ou dépassé la limite d'alerte inférieure. (valeur par défaut : FALSE)
State	OUT	Int Mode de fonctionnement en cours du régulateur PID. (valeur par défaut : 0) Vous pouvez modifier l'état de fonctionnement à l'aide du paramètre d'entrée Mode et un front montant pour ModeActivate: <ul style="list-style-type: none">• State = 0 : Inactif• State = 1 : optimisation préalable• State = 2 : Optimisation fine manuelle• State = 3 : Mode automatique• State = 4 : Mode manuel• State = 5 : Accostage de la valeur de réglage de remplacement• State = 6 : Mesure du temps de transition• State = 7 : Surveillance d'erreur• State = 8 : Accostage de la valeur de réglage de remplacement avec surveillance d'erreur• State = 10 : Mode manuel sans signaux de butée d'extrémité

Paramètre et type		Type de données	Description
Error	OUT	Bool	<p>Si Error = TRUE, au moins un message d'erreur est en attente. (valeur par défaut : FALSE)</p> <p>Note : Le paramètre Error dans V1.x PID était le champ ErrorBits qui contenait les codes d'erreurs. C'est maintenant une signalisation booleenne qui indique qu'une erreur est survenue.</p>
ErrorBits	OUT	DWord	<p>L'instruction PID_3StepErrorBits table de paramètres (Page 216) définit les messages d'erreurs qui sont en attente. (valeur par défaut : DW#16#0000 (aucune erreur)). ErrorBits est rémanent et réinitialisé sur un front montant pour Reset ou ErrorAck.</p> <p>Note : Dans V1.x, le paramètre ErrorBits a été défini comme le paramètre Error et n'existe pas.</p>

Fonctionnement du régulateur PID_3Step

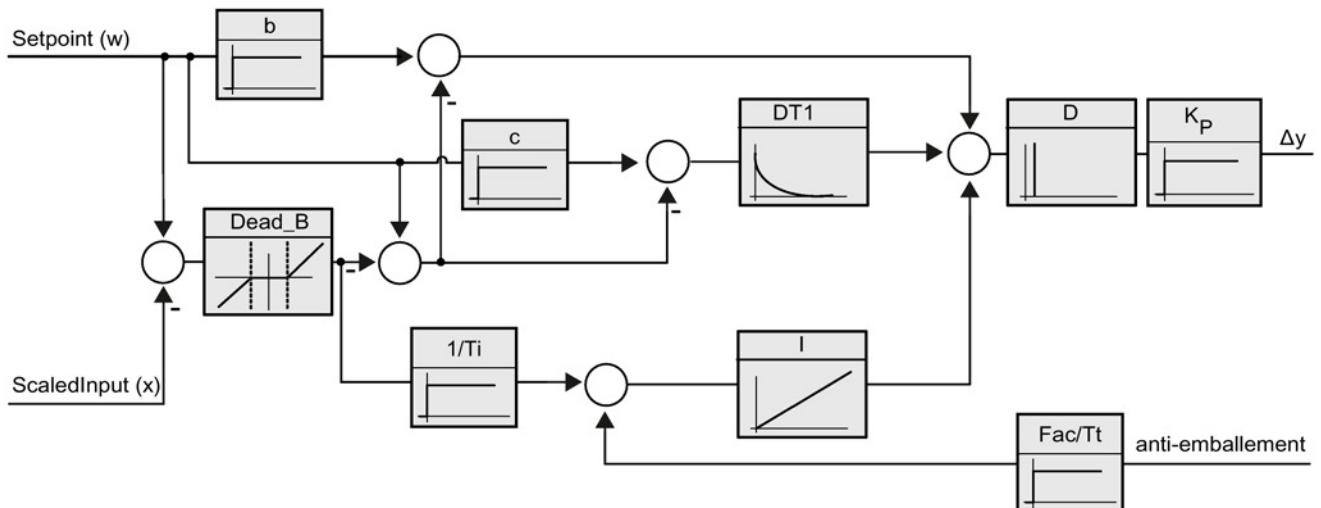


Figure 8-3 Fonctionnement du régulateur PID_3Step en tant que régulateur PID T1 avec anti-emballement

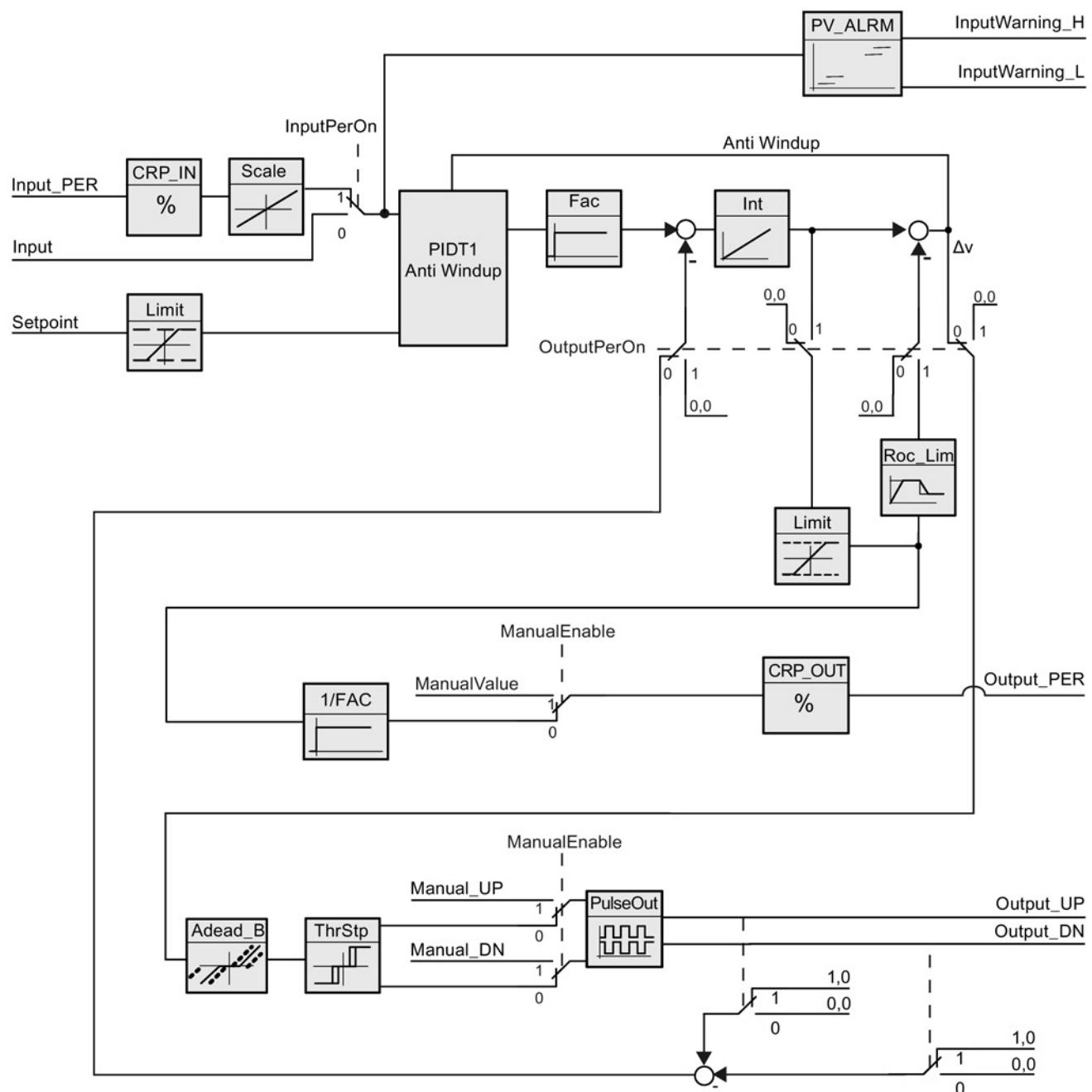


Figure 8-4 Fonctionnement du régulateur PID_3Step sans signalisation de position

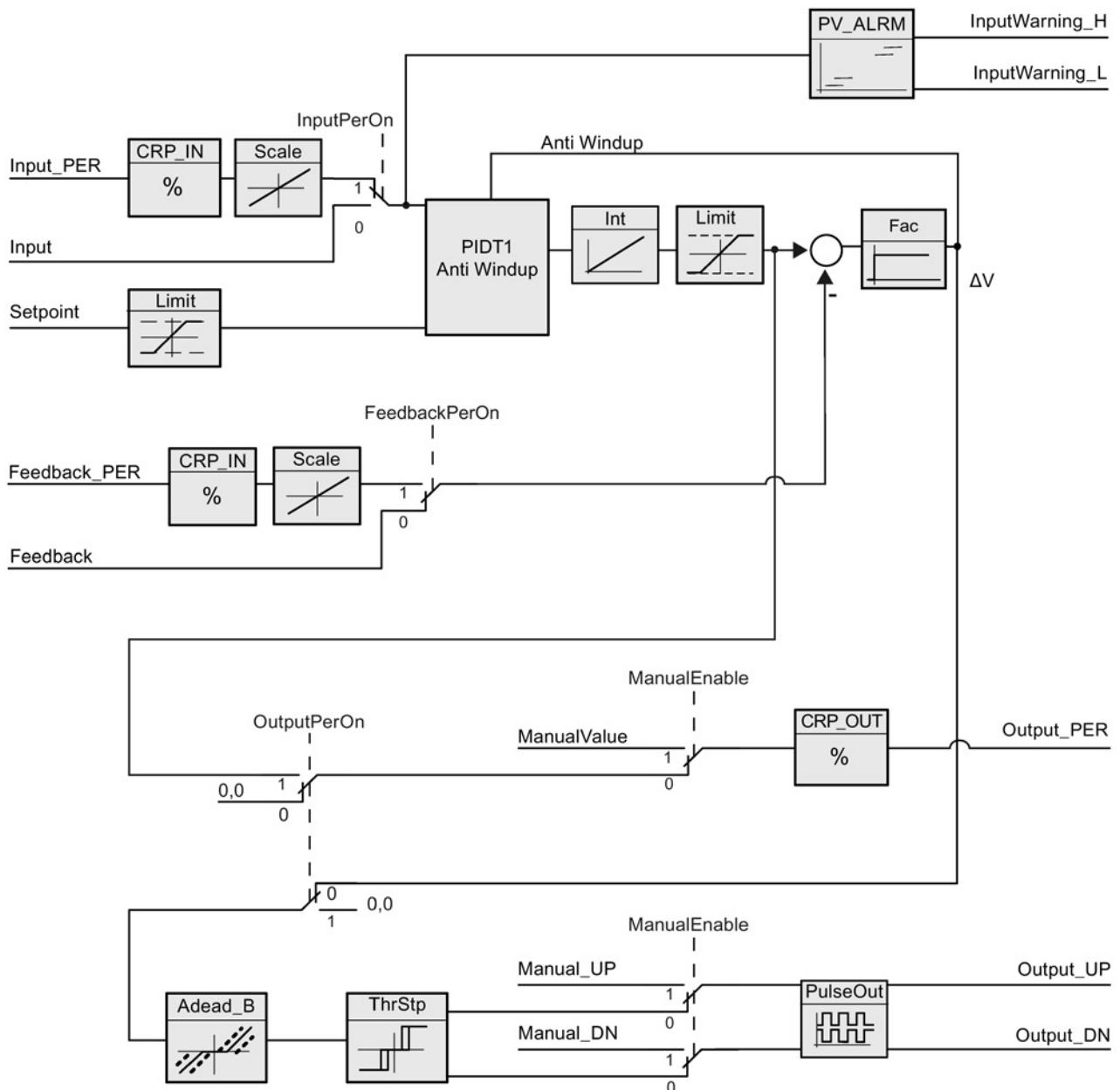


Figure 8-5 Fonctionnement du régulateur PID_3Step avec signalisation de position activée

8.5

Paramètres ErrorBit de l'instruction PID_3Step

Si plusieurs erreurs sont en attente, les valeurs des codes d'erreur sont données au moyen d'une addition binaire. Ainsi, l'affichage du code d'erreur 0003 signifie que les erreurs 0001 et 0002 sont également en attente.

Tableau 8- 8 Paramètres ErrorBit de l'instruction PID_3STEP

ErrorBit (DW#16#...)	Description
0000	Pas d'erreur
0001 ^{1,2}	Le paramètre Input se situe en dehors des limites de la mesure. Input > Config.InputUpperLimit Input < Config.InputLowerLimit
0002 ^{2,3}	Valeur invalide pour le paramètre Input_PER. Vérifiez s'il y a une erreur en attente au niveau de l'entrée analogique.
0004 ⁴	Erreur pendant l'optimisation fine. L'oscillation de la mesure n'a pas pu être conservée.
0010 ⁴	La consigne a été modifiée pendant l'optimisation. Note : Vous pouvez définir la fluctuation permise sur la consigne à la variable CancelTuningLevel.
0020	L'optimisation préalable n'est pas permise pendant l'optimisation fine. Note : Si ActivateRecoverMode à la valeur TRUE avant que l'erreur ne survienne, PID_3Step reste en mode optimisation fine.
0080 ⁴	Erreur pendant l'optimisation préalable. Configuration incorrecte des limites de la valeur de réglage. Vérifiez si les limites de la valeur de réglage sont configurées correctement et correspondent à la logique du régulateur.
0100 ⁴	Une erreur pendant l'optimisation fine a entraîné des paramètres invalides.
0200 ^{2,3}	Valeur invalide pour le paramètre Input. La valeur a un format numérique invalide.
0400 ^{2,3}	Le calcul de la valeur de réglage a échoué. Vérifiez les paramètres PID.
0800 ^{1,2}	Erreur de période d'échantillonnage : PID_3Step n'est pas appelé dans la période d'échantillonnage de l'OB d'alarme cyclique.
1000 ^{2,3}	Valeur invalide pour le paramètre Setpoint. La valeur a un format numérique invalide.
2000 ^{1,2,5}	Valeur invalide pour le paramètre Feedback_PER. Vérifiez s'il y a une erreur en attente au niveau de l'entrée analogique.
4000 ^{1,2,5}	Valeur invalide pour le paramètre Feedback. La valeur a un format numérique invalide.
8000 ^{1,2}	Erreur pendant la signalisation de position numérique. Actuator_H = TRUE et Actuator_L = TRUE. L'actionneur ne peut pas être déplacé vers la valeur de réglage de remplacement et reste dans la position actuelle. Le mode manuel n'est pas possible dans cet état. Afin de déplacer l'actionneur de cet état, vous devez désactiver la "butée d'extrême de l'actionneur" (Config.ActuatorEndStopOn à la valeur FALSE) ou passer sur le mode manuel sans signaux de butée d'extrême (Mode = 10).

ErrorBit (DW#16#...)	Description
10000	<p>Valeur invalide pour le paramètre ManualValue. La valeur a un format numérique invalide.</p> <p>L'actionneur ne peut pas être déplacé vers la valeur manuelle et reste dans la position en cours.</p> <p>Affecte une valeur valide dans ManualValue ou déplace l'actionneur en mode manuel avec Manual_UP et Manual_DN.</p>
20000	<p>Valeur invalide pour la variable SavePosition : La valeur a un format numérique invalide.</p> <p>L'actionneur ne peut pas être déplacé vers la valeur de réglage de remplacement et reste dans la position actuelle.</p>
40000	<p>Valeur invalide pour le paramètre Disturbance. La valeur a un format numérique invalide.</p> <p>Note : Si le mode automatique était actif et si ActivateRecoverMode a la valeur FALSE avant l'apparition de l'erreur, Disturbance est défini sur zéro. PID_3Step reste en mode automatique.</p> <p>Note : Si l'optimisation préalable ou l'optimisation fine était active et ActivateRecoverMode a la valeur TRUE avant que l'erreur ne se produise, PID_3Step passe dans l'état de fonctionnement qui est enregistré pour le paramètre Mode. Si une perturbation dans la phase actuelle n'a pas d'effet sur la valeur de réglage, l'optimisation n'est pas annulée.</p> <p>L'erreur n'a pas d'effet pendant la mesure du temps de transition.</p>

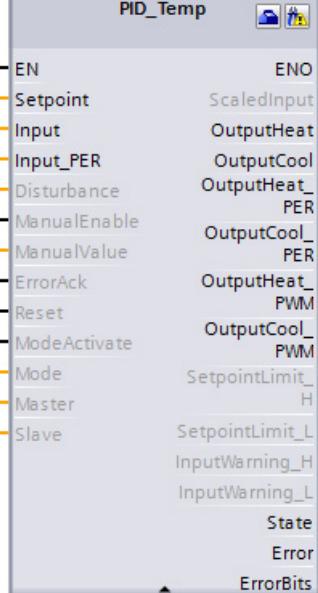
- ¹ Note : Si l'état automatique était actif avant que l'erreur ne se produise et ActivateRecoverMode a la valeur TRUE, PID_3Step reste en mode automatique.
- ² Note : Si le mode optimisation préalable, optimisation fine ou mesure du temps de transition était actif et si ActivateRecoverMode = TRUE avant que l'erreur ne survienne, PID_3Step passe sur l'état de fonctionnement qui a été enregistré pour le paramètre Mode.
- ³ Note : Si le mode automatique était actif avant que l'erreur ne survienne et si ActivateRecoverMode = TRUE, PID_3Step passe dans l'état "Accostage de la valeur de réglage de remplacement avec une surveillance d'erreur" ou "Surveillance d'erreur". Dès que l'erreur n'est plus en attente, PID_3Step repasse en mode automatique.
- ⁴ Note : Si ActivateRecoverMode a la valeur TRUE avant que l'erreur ne se produise, PID_3Step annule l'optimisation et passe dans l'état de fonctionnement qui est enregistré pour le paramètre Mode.
- ⁵ L'actionneur ne peut pas être déplacé vers la valeur de réglage de remplacement et reste dans la position actuelle. En mode manuel, vous pouvez modifier la position de l'actionneur uniquement avec Manual_UP et Manual_DN, et pas avec ManualValue.

8.6 Instruction PID_Temp

8.6.1 Présentation

L'instruction PID_Temp offre un régulateur PID universel qui permet de gérer des exigences spécifiques en matière de régulation de la température.

Tableau 8- 9 Instruction PID_Temp

CONT/LOG	SCL	Description
	<pre> "PID_Temp_1"(Setpoint:=_real_in_, Input:=_real_in_, Input_PER:=_int_in_, Disturbance:=_real_in_, ManualEnable:=_bool_in_, ManualValue:=_real_in_, ErrorAck:=_bool_in_, Reset:=_bool_in_, ModeActivate:=_bool_in_, Mode:=_int_in_, Master:=_dword_in Save:=_dword_in ScaledInput=>_real_out_, OutputHeat=>_real_out_, OutputCool=>_real_out_, OutputHeat_PER=>_int_out_, OutputCool_PER=>_int_out_, Out- putHeat_PWM=>_bool_out_, Out- putCool_PWM=>_bool_out_, SetpointLimit_H=>_bool_out_, SetpointLimit_L=>_bool_out_, InputWarning_H=>_bool_out_, InputWarning_L=>_bool_out_, State=>_int_out_, Error=>_bool_out_, ErrorBits=>_dword_out_); </pre>	<p>PID_Temp offre les possibilités suivantes :</p> <ul style="list-style-type: none"> • Chauffage et refroidissement du processus grâce à différents actionneurs • Réglage automatique intégré pour gérer les processus de température • Effet de cascade pour traiter plusieurs températures qui dépendent du même actionneur

¹ STEP 7 crée automatiquement l'objet technologique et le DB d'instance lorsque vous insérez l'instruction. Le DB d'instance contient les paramètres de l'objet technologique.

² Dans l'exemple SCL, "PID_Temp_1" est le nom du DB d'instance.

Tableau 8- 10 Types de données pour les paramètres

Paramètre et type		Type de données	Description
Setpoint	IN	Real	Consigne du régulateur PID en mode automatique. (valeur par défaut : 0.0)
Input	IN	Real	Une variable du programme utilisateur est utilisée comme source pour la mesure. (valeur par défaut : 0.0) Si vous utilisez le paramètre Input alors il faut définir Config.InputPerOn = FALSE.
Input_PER	IN	Int	Une entrée analogique est utilisée comme source pour la mesure. (valeur par défaut : 0) Si vous utilisez le paramètre Input_PER alors il faut définir Config.InputPerOn = TRUE.
Disturbance	IN	Real	Variable de perturbation ou valeur de pré-régulation
ManualEnable	IN	Bool	Active ou désactive le mode de fonctionnement manuel. (valeur par défaut : FALSE): <ul style="list-style-type: none"> • Un front de FALSE à TRUE active le mode manuel, tant que Etat = 4, le mode reste inchangé. • Tant que ManualEnable est VRAI, vous ne pouvez pas modifier l'état de fonctionnement à l'aide d'un front montant pour ModeActivate ou utiliser la boîte de dialogue de mise en service. • Un front TRUE à FALSE active l'état de fonctionnement qui est affecté par Mode. Note : Nous vous recommandons de modifier l'état de fonctionnement à l'aide de ModeActivate uniquement.
ManualValue	IN	Real	Valeur de sortie pour le mode manuel. (valeur par défaut : 0.0) Vous pouvez utiliser des valeurs à partir Config.OutputLowerLimit à Config.OutputUpperLimit.
ErrorAck	IN	Bool	Réinitialise les ErrorBits et les messages d'alerte avec un front de FALSE à TRUE. (valeur par défaut : FALSE)
Reset	IN	Bool	Redémarre le régulateur. (valeur par défaut : FALSE): <ul style="list-style-type: none"> • Front de FALSE à TRUE : <ul style="list-style-type: none"> – Passe en mode "inactif" – Réinitialise les ErrorBits et les sorties d'alerte – Efface l'action intégrale – Maintient les paramètres PID • Tant que Reset à la valeur TRUE, PID_Temp reste en mode Inactif (Etat = 0). • Front de TRUE à FALSE : <ul style="list-style-type: none"> – PID_Temp passe à l'état de fonctionnement qui est enregistré dans le paramètre Mode.
ModeActivate	IN	Bool	Le PID_Temp passe à l'état de fonctionnement qui est enregistré dans le paramètre Mode avec un front de FALSE à TRUE. (valeur par défaut : FALSE)

Paramètre et type		Type de données	Description
Mode	IN/OUT	Int	<p>Activé sur le front montant de l'entrée Mode Activate.</p> <p>Sélection du mode de fonctionnement (valeur par défaut : 0.0):</p> <ul style="list-style-type: none"> • Mode = 0 : Inactif • Mode = 1 : Optimisation préalable • Mode = 2 : Optimisation fine • Mode = 3 : Mode automatique • Mode = 4 : Mode manuel <p>"Valeur de réglage de remplacement avec surveillance d'erreur" (State = 5). Celui-ci ne peut pas être activé par l'utilisateur ; il ne s'agit que d'une réaction automatique à une erreur.</p>
Master	IN/OUT	DWord	<p>Connexion en cascade au maître (anti-emballement et conditions d'optimisation). (valeur par défaut : DW#16#0000)</p> <ul style="list-style-type: none"> • Bits 0 - 15 : Pas utilisés dans l'instruction PID_Temp • Bits 16 - 23 : Compteur de limite : Un esclave incrémente cette valeur si elle atteint sa limite. Le nombre d'esclaves en limitation est traité pour la fonctionnalité anti-emballement (Voir le paramètre Config.Cascade.AntiWindUpMode). • Bit 24 : IsAutomatic : Ce bit est défini sur "1" si tous les esclaves de ce contrôleur sont en mode automatique et sont traités pour vérifier les conditions d'optimisation en cascade. Ce bit est identique au paramètre AllSlaveAutomaticState. • Bit 25 : "IsReplacement-Setpoint" : Ce bit est défini sur "1" si un esclave de ce contrôleur a la "Consigne de remplacement" activée et est traité pour vérifier les conditions d'optimisation en cascade. La valeur inversée est sauvegardée dans le paramètre NoSlaveReplacementSetpoint.
ScaledInput	OUT	Real	Mesure mise à l'échelle. (valeur par défaut : 0.0)
OutputHeat ¹	OUT	Real	<p>Valeur de sortie pour le chauffage en mode REEL. (valeur par défaut : 0.0)</p> <p>Cette valeur de sortie est calculée, indépendamment de la sélection de sortie, à l'aide du paramètre Config.Output.Heat.Select.</p>
OutputCool ¹	OUT	Real	<p>Valeur de sortie pour le refroidissement en mode REEL. (valeur par défaut : 0.0)</p> <p>Cette valeur de sortie est calculée, indépendamment de la sélection de sortie, à l'aide du paramètre Config.Output.Cool.Select.</p>
OutputHeat_PER ¹	OUT	Int	<p>Valeur de sortie pour le chauffage au format périphérique (valeur par défaut : 0)</p> <p>Cette valeur de sortie n'est calculée que si elle est sélectionnée à l'aide du paramètre Config.Output.Heat.Select = 2. Si elle n'est pas sélectionnée, cette sortie est toujours égale à "0".</p>
OutputCool_PER ¹	OUT	Int	<p>Valeur de sortie pour le refroidissement au format périphérique (valeur par défaut : 0)</p> <p>Cette valeur de sortie n'est calculée que si elle est sélectionnée à l'aide du paramètre Config.Output.Cool.Select = 2. Si elle n'est pas sélectionnée, cette sortie est toujours égale à "0".</p>

Paramètre et type		Type de données	Description
OutputHeat_PWM1	OUT	Bool	<p>Valeur de sortie à modulation de largeur des impulsions pour le chauffage. (valeur par défaut : FALSE)</p> <p>Cette valeur de sortie n'est calculée que si elle est sélectionnée à l'aide du paramètre Config.Output.Heat.Select = 1 (valeur par défaut). Si elle n'est pas sélectionnée, cette sortie est toujours FALSE.</p>
OutputCool_PWM1	OUT	Bool	<p>Valeur de sortie à modulation de largeur des impulsions pour le refroidissement. (valeur par défaut : FALSE)</p> <p>Cette valeur de sortie n'est calculée que si elle est sélectionnée à l'aide du paramètre Config.Output.Cool.Select = 1 (valeur par défaut). Si elle n'est pas sélectionnée, cette sortie est toujours FALSE.</p>
SetpointLimit_H	OUT	Bool	<p>Limite supérieure de la consigne. (valeur par défaut : FALSE)</p> <p>Si SetpointLimit_H a la valeur TRUE, la limite supérieure absolue de la consigne est atteinte.(Setpoint \geq Config.SetpointUpperLimit).</p> <p>La consigne est limitée à Config.SetpointUpperLimit.</p>
SetpointLimit_L	OUT	Bool	<p>Limite inférieure de la consigne. (valeur par défaut : FALSE)</p> <p>Si SetpointLimit_L a la valeur TRUE, la limite inférieure absolue de la consigne est atteinte.(Setpoint \leq Config.SetpointLowerLimit).</p> <p>La consigne est limitée à Config.SetpointLowerLimit.</p>
InputWarning_H	OUT	Bool	<p>Si InputWarning_H a la valeur TRUE, la mesure a atteint ou dépassé la limite d'alerte supérieure. (valeur par défaut : FALSE)</p>
InputWarning_L	OUT	Bool	<p>Si InputWarning_L a la valeur TRUE, la mesure de processus a atteint ou est tombée en dessous de la limite d'alerte inférieure. (valeur par défaut : FALSE)</p>
State	OUT	Int	<p>Mode de fonctionnement en cours du régulateur PID. (valeur par défaut : 0)</p> <p>Vous pouvez modifier l'état de fonctionnement à l'aide du paramètre d'entrée Mode et un front montant pour ModeActivate :</p> <ul style="list-style-type: none"> • State = 0 : Inactif • State = 1 : optimisation préalable • State = 2: Optimisation fine • State = 3 : Mode automatique • State = 4 : Mode manuel • State = 5 : Valeur de réglage de remplacement avec surveillance d'erreur
Error	OUT	Bool	<p>Si Error a la valeur TRUE, au moins un message d'erreur est en attente dans ce cycle. (valeur par défaut : FALSE)</p> <p>Note : Le paramètre Error dans V1.x PID était le champ ErrorBits qui contenait les codes d'erreurs. C'est maintenant une signalisation booléenne qui indique qu'une erreur est survenue.</p>
ErrorBits	OUT	DWord	<p>Le tableau des paramètres ErrorBits (Page 229) de l'instruction PID_Temp présente les messages d'erreur qui sont en attente. (valeur par défaut : DW#16#0000 (aucune erreur)). ErrorBits est rémanent et réinitialisé sur un front montant pour Reset ou ErrorAck.</p> <p>Note : Dans V1.x, le paramètre ErrorBits a été défini comme le paramètre Error et n'existe pas.</p>

Paramètre et type		Type de données	Description
Warning	OUT	DWord	Le tableau des paramètres Warning de l'instruction PID_Temp présente les messages d'alerte importants pour l'utilisateur, qui sont en attente. (valeur par défaut : DW#16#0000 (pas d'alerte)).
WarningInternal	OUT	DWord	Le tableau des paramètres WarningInternal de l'instruction PID_Temp présente les messages d'alerte internes qui sont en attente (comprend toutes les alertes). (valeur par défaut : DW#16#0000 (pas d'alerte interne)).

¹ Vous pouvez utiliser les sorties de la Output, Output_PER et Output_PWM des paramètres en parallèle.

8.6.2 Fonctionnement du régulateur PID_Temp

Sélection de la commande de chauffage et/ou de refroidissement

Vous devez d'abord choisir au paramètre "ActivateCooling" si vous avez besoin d'un dispositif de refroidissement en plus de la sortie de chauffage. Ensuite, vous devez indiquer au paramètre "AdvancedCooling" si vous voulez utiliser deux jeux de paramètres PID (mode avancé) ou un seul jeu de paramètres PID avec un facteur de chauffage/refroidissement supplémentaire.

Utilisation du facteur de refroidissement (CoolFactor)

Si vous voulez appliquer un facteur de chauffage/refroidissement, vous devez définir la valeur manuellement. Vous devez identifier cette valeur grâce aux données techniques de votre application (rapport entre le gain proportionnel des actionneurs (par exemple, le rapport entre la puissance de chauffage et la puissance de refroidissement maximum des actionneurs)) et l'affecter au paramètre "CoolFactor". Un facteur de chauffage/refroidissement de 2,0 signifie que le dispositif de chauffage est deux fois plus puissant que le dispositif de refroidissement. Si vous utilisez un facteur de refroidissement, PID_Temp calcule le signal de sortie, et selon son signe, il multiplie le signal de sortie par le facteur de chauffage/refroidissement (lorsque le signe est négatif), ou non (lorsque le signe est positif).

Utilisation de deux jeux de paramètres PID

Différents jeux de paramètres PID pour le chauffage et le refroidissement peuvent être automatiquement détectés pendant la mise en service. Vous pouvez escompter une meilleure performance de la commande, en comparaison du facteur de chauffage/refroidissement car, en plus de différents gains proportionnels, vous pouvez envisager différentes temporisations avec deux jeux de paramètres. Toutefois, l'inconvénient de cette méthode est que le processus d'optimisation peut durer plus longtemps. Si la commutation des paramètres PID est activée (Config.AdvancedCooling = TRUE), le régulateur PID_Temp détecte en "mode automatique" (le contrôle est actif) s'il est nécessaire de chauffer ou de refroidir à ce moment et il utilise les jeux de paramètres PID pour la commande.

ControlZone

Le régulateur PID_Temp vous permet de définir une zone de régulation pour chaque jeu de paramètres au paramètre "ControlZone". Si l'écart de régulation (consigne - mesure) se trouve dans la zone de régulation, PID_Temp utilise l'algorithme PID pour calculer les signaux de sortie. Cependant, si l'écart de régulation dépasse la plage définie, la sortie est mise à la valeur de réglage de chauffage ou de refroidissement maximum (si la sortie de refroidissement est activée) / à la valeur de réglage de chauffage minimum (si la sortie de refroidissement est désactivée). Vous pouvez utiliser cette fonction pour atteindre la consigne souhaitée plus rapidement, notamment pour le chauffage initial dans le cadre de processus thermiques lents.

Zone morte (DeadZone)

Dans le paramètre "DeadZone", vous pouvez définir un intervalle d'écart de régulation pour le chauffage et le refroidissement négligé par l'algorithme PID. Cela signifie que l'écart de régulation est supprimé dans cette plage et que le régulateur PID_Temp réagit comme si la consigne et les valeurs de process étaient identiques. Ainsi, vous pouvez éviter les interventions inutiles du régulateur autour de la consigne et ménager l'actionneur. Si vous voulez appliquer une zone morte, vous devez définir la valeur manuellement. L'auto-optimisation ne fixe pas automatiquement de valeur de zone morte. La zone morte est symétrique (entre -Retain.CtrlParams.Heat.DeadZone et +Retain.CtrlParams.Heat.DeadZone) pour les régulateurs de chauffage sans refroidissement ou les régulateurs de chauffage/refroidissement employant un facteur de refroidissement. La zone morte peut être asymétrique (entre -Retain.CtrlParams.Cool.DeadZone et +Retain.CtrlParams.Heat.DeadZone) pour les régulateurs de chauffage/refroidissement employant deux jeux de paramètres PID.

Fonctionnement du régulateur PID_Temp

Les schémas fonctionnels suivants illustrent le fonctionnement standard et en cascade de l'instruction PID_Temp :

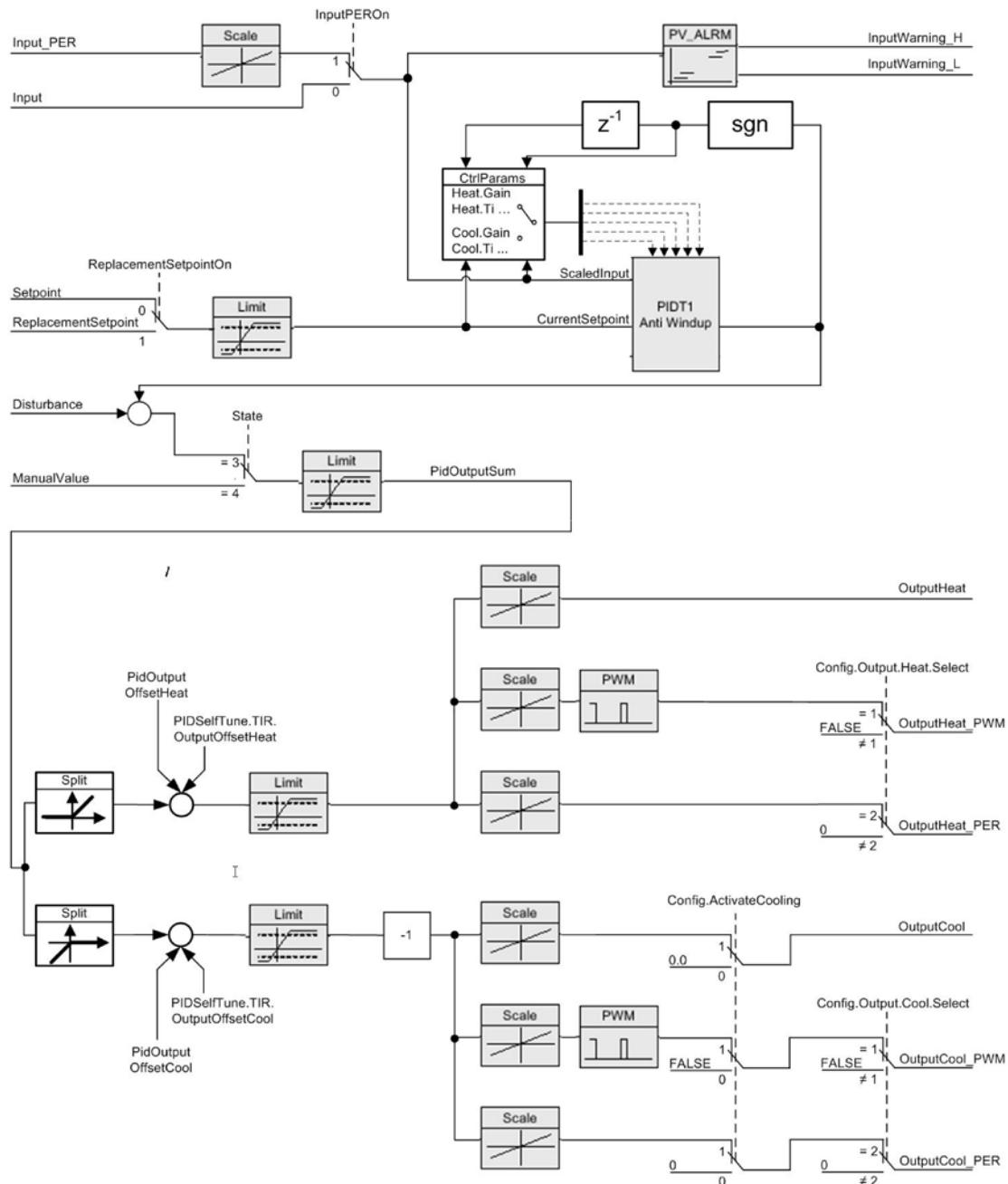


Figure 8-6 PID_Temp_Operation_Block_Diagram

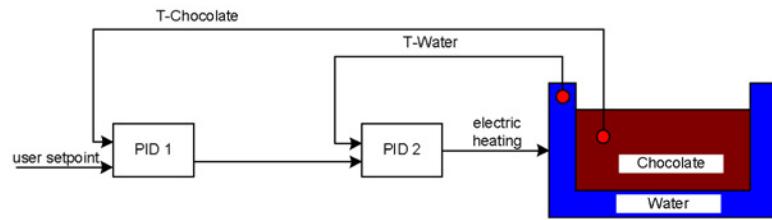


Figure 8-7 PID_Temp_Cascade_Operation_Block_Diagram

8.6.3 Régulateurs en cascade

Vous pouvez mettre en cascade des régulateurs thermiques PID pour traiter plusieurs températures qui dépendent du même actionneur.

Ordre d'appel

Vous devez appeler des régulateurs PID mis en cascade dans le même cycle d'OB. D'abord, il vous faut appeler le maître, puis le ou les esclave(s) suivant(s) dans le flux des signaux de commande pour finir au dernier esclave de la cascade. L'instruction PID_Temp n'effectue pas de vérification automatique de l'ordre d'appel.

Liaisons de communication

Lorsque vous mettez en cascade des régulateurs, vous devez connecter le maître à l'esclave de telle sorte qu'ils puissent partager des informations. Vous devez connecter le paramètre IN/OUT "maître" d'un esclave au paramètre IN/OUT "esclave" de son maître dans le sens du flux des signaux.

La figure ci-dessous représente le montage de régulateurs PID_Temp en cascade avec deux sous-cascades : "PID_Temp1" fournit la consigne. La configuration connecte les sorties de "PID_Temp2", "PID_Temp3", "PID_Temp5", "PID_Temp6", et "PID_Temp8" au processus :

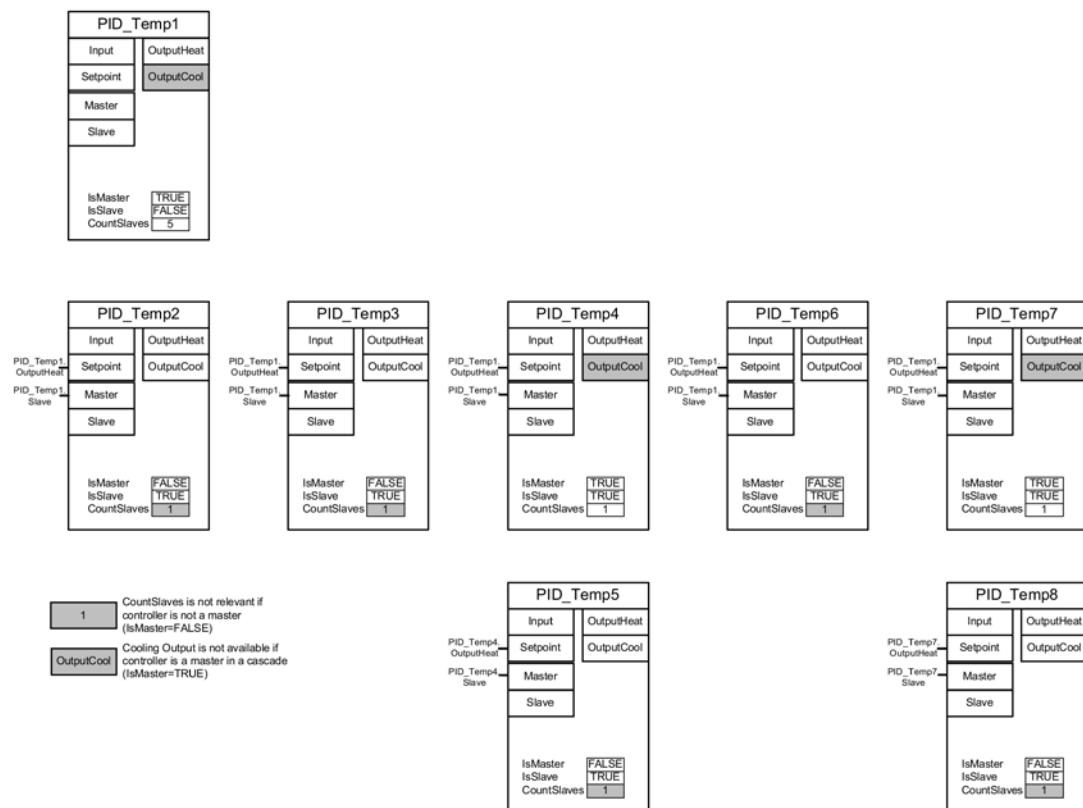


Figure 8-8 PID_Temp_Cascading_communication_connection

Consigne de remplacement

L'instruction PID_Temp fournit une deuxième entrée de consigne au paramètre "ReplacementSetpoint" que vous pouvez activer en réglant le paramètre "ReplacementSetpointOn" = TRUE. Vous pouvez utiliser "ReplacementSetpoint" comme votre entrée de consigne pendant la mise en service ou l'optimisation d'un régulateur esclave sans être obligé de couper la connexion mesure-consigne entre le maître et l'esclave. Cette connexion est nécessaire au fonctionnement normal de la cascade.

Par conséquent, vous n'êtes pas obligé de modifier votre programme et de le télécharger si vous voulez séparer temporairement un esclave de son maître. Vous devez seulement activer "ReplacementSetpoint" et la désactiver ensuite lorsque vous avez terminé. La consigne s'applique à l'algorithme PID lorsque vous pouvez voir sa valeur au paramètre "CurrentSetpoint".

Auto-optimisation

L'auto-optimisation d'un régulateur maître mis en cascade doit remplir les conditions suivantes :

- La mise en service doit être effectuée à partir de l'esclave interne vers le premier maître.
- Tous les esclaves du maître doivent être en "mode automatique".
- La sortie du maître doit servir de consigne aux esclaves.

L'instruction PID_Temp apportera l'aide suivante pour l'auto-optimisation dans la cascade :

- Si vous démarrez l'auto-optimisation pour un régulateur maître, celui-ci vérifie que tous les esclaves sont en "mode automatique" et vérifie la désactivation de la fonction consigne de remplacement pour tous les esclaves ("ReplacementSetpointOn" = FALSE). Si ces conditions ne sont pas remplies, vous ne pouvez pas auto-optimiser le maître. Le maître annule l'optimisation, passe en "mode inactif" (si "ActivateRecoverMode" = FALSE) ou revient dans le mode enregistré dans le paramètre "Mode" (si "ActivateRecoverMode" = TRUE). Le maître affiche le message d'erreur 200000hex ("Erreur dans un maître de la cascade. Les esclaves ne sont pas en mode automatique ou ont une consigne de remplacement activée, et empêchent l'optimisation du maître").
- Lorsque tous les esclaves sont en "mode automatique", le système règle le paramètre "AllSlaveAutomaticState" sur TRUE. Vous pouvez appliquer ce paramètre dans vos programmes ou localiser la cause de l'erreur 200000hex.
- Lorsque "ReplacementSetpoint" est désactivé pour tous les esclaves, le système règle le paramètre "NoSlaveReplacementSetpoint" sur TRUE. Vous pouvez appliquer ce paramètre dans leurs programmes ou localiser la cause de l'erreur 200000hex.

Utilisez la boîte de dialogue de mise en service de l'instruction PID_Temp pour bénéficier d'une assistance supplémentaire pour l'optimisation de la cascade (Page 251).

Etats de fonctionnement et gestion des erreurs

Le régulateur PID_Temp n'autorise pas son maître ou ses esclaves à commuter l'état de fonctionnement. Cela signifie qu'un maître d'une cascade reste dans son état actuel lorsqu'un esclave signale une erreur. C'est un avantage si deux esclaves parallèles ou plus fonctionnent avec ce régulateur maître ; une erreur dans un segment n'entraîne pas l'arrêt du segment parallèle.

De même, un esclave d'une cascade reste dans son état de fonctionnement actuel si son maître a une erreur. Toutefois, le fonctionnement ultérieur de l'esclave dépend de la configuration du maître car la consigne de l'esclave correspond à la sortie du maître. Par conséquent, si vous avez configuré "ActivateRecoverMode" = TRUE sur le maître et qu'une erreur survient, le maître émet la dernière valeur de réglage valide ou une valeur de réglage de remplacement comme consigne pour l'esclave. Si vous avez configuré "ActivateRecoverMode" = FALSE sur le maître, le maître passe en "mode inactif" et met toutes les sorties à "0,0" de telle sorte que l'esclave utilise "0,0" comme consigne.

C'est parce que seuls les régulateurs esclaves ont un accès direct aux actionneurs et que ces derniers restent dans leur état de fonctionnement en cas d'erreur d'un maître, que vous pouvez éviter des dommages au processus. Par exemple, pour les appareils de traitement du plastique, il est catastrophique que les esclaves cessent de travailler, arrêtent les actionneurs et permettent au plastique de durcir dans l'appareil uniquement parce que le régulateur maître a subi une erreur.

Anti-emballement

Un esclave d'une cascade reçoit sa consigne de la sortie de son maître. Si l'esclave atteint ses propres limites de réglage alors que le maître détecte toujours un écart de régulation (consigne - mesure), le maître bloque ou réduit sa contribution d'intégration afin d'éviter le phénomène dit d'emballement. En cas d'emballement, le maître augmente considérablement sa contribution d'intégration et il doit commencer par la réduire avant que le régulateur puisse retrouver une réaction normale. Un "emballement" de ce type nuit à la dynamique de la commande. PID_Temp fournit des moyens d'éviter cet effet dans une cascade par la configuration du paramètre Config.Cascade.AntiWindUpMode" du régulateur maître :

Valeur	Description
0	Désactive la fonction anti-emballement.
1	Réduit la contribution d'intégration du régulateur maître au rapport entre "esclaves en limitation" et "esclaves existants" (paramètre "CountSlaves").
2	Bloque la contribution d'intégration du maître dès qu'un esclave atteint sa limite. N'est pertinent que si "Config.Cascade.IsMaster" = TRUE.

8.7

Paramètres ErrorBit de l'instruction PID_Temp

Si le régulateur PID a plusieurs alertes en attente, il affiche les valeurs des codes d'erreur au moyen d'une addition binaire. Ainsi, l'affichage du code d'erreur 0003 signifie, par exemple, que les erreurs 0001 et 0002 sont en attente.

Tableau 8- 11 Paramètres ErrorBit de l'instruction PID_Temp

ErrorBit (DW#16#...)	Description
0000	Pas d'erreur
0001 ^{1, 2}	Le paramètre Input se situe en dehors des limites de la mesure. Input > Config.InputUpperLimit Input < Config.InputLowerLimit
0002 ^{2, 3}	Valeur invalide pour le paramètre Input_PER. Vérifiez s'il y a une erreur en attente au niveau de l'entrée analogique.
0004 ⁴	Erreur pendant l'optimisation fine. L'oscillation de la mesure n'a pas pu être conservée.
0008 ⁴	Erreur au démarrage de l'optimisation préalable. La mesure est trop proche de la consigne. Démarrer l'optimisation fine.
0010 ⁴	La consigne a été modifiée pendant l'optimisation. Note : Vous pouvez définir la fluctuation permise sur la consigne à la variable CancelTuningLevel.
0020	L'optimisation préalable n'est pas permise pendant l'optimisation fine. Note : Si ActivateRecoverMode à la valeur TRUE avant que l'erreur ne survienne, PID_Temp reste en mode optimisation fine.
0040 ⁴	Erreur pendant l'optimisation préalable. Le refroidissement n'a pas pu réduire la valeur de process.
0080 ⁴	Erreur pendant l'optimisation préalable. Configuration incorrecte des limites de la valeur de réglage. Vérifiez si les limites de la valeur de réglage sont configurées correctement et correspondent à la logique du régulateur.
0100 ⁴	Une erreur pendant l'optimisation fine a entraîné des paramètres invalides.
0200 ^{2, 3}	Valeur invalide pour le paramètre Input. La valeur a un format numérique invalide.
0400 ^{2, 3}	Le calcul de la valeur de réglage a échoué. Vérifiez les paramètres PID.
0800 ^{1, 2}	Erreur de période d'échantillonnage : PID_Temp n'est pas appelé dans la période d'échantillonnage de l'OB d'alarme cyclique.
1000 ^{2, 3}	Valeur invalide pour le paramètre Setpoint. La valeur a un format numérique invalide.
10000	Valeur invalide pour le paramètre ManualValue. La valeur a un format numérique invalide. Note : Si ActivateRecoverMode à la valeur TRUE avant que l'erreur ne survienne, PID_Temp utilise SubstituteOutput comme valeur de réglage. Dès que vous affectez une valeur invalide dans le paramètre ManualValue, PID_Temp l'utilise comme valeur de réglage.

ErrorBit (DW#16#...)	Description
20000	<p>Valeur invalide pour la variable SubstituteValue : La valeur a un format numérique invalide.</p> <p>PID_Temp utilise la limite inférieure de la valeur de réglage comme valeur de réglage.</p> <p>Remarque : Si le mode automatique était actif avant l'apparition de l'erreur, ActivateRecoverMode a la valeur TRUE et l'erreur n'est plus en attente, PID_Temp repasse en mode automatique.</p>
40000	<p>Valeur invalide pour le paramètre Disturbance. La valeur a un format numérique invalide.</p> <p>Note : Si le mode automatique était actif et si ActivateRecoverMode a la valeur FALSE avant l'apparition de l'erreur, Disturbance est défini sur zéro. PID_Temp reste en mode automatique.</p> <p>Remarque : Si l'optimisation préalable ou l'optimisation fine était active et ActivateRecoverMode a la valeur TRUE avant que l'erreur ne se produise, PID_Temp passe à l'état de fonctionnement qui est enregistré pour le paramètre Mode. Si une perturbation dans la phase actuelle n'a pas d'effet sur la valeur de réglage, l'optimisation n'est pas annulée.</p>
200000	Erreur dans un maître de la cascade. Les esclaves ne sont pas en mode automatique ou ont une consigne de remplacement activée, ce qui empêche l'optimisation du maître.
400000	Le régulateur PID ne permet pas l'optimisation préalable pour le chauffage lorsque le refroidissement est actif.
800000	La valeur de process doit être proche de la consigne pour démarrer l'optimisation préalable pour le refroidissement.
1000000	Erreur au démarrage de l'optimisation. "Heat.EnableTuning" et "Cool.EnableTuning" ne sont pas mis à 1 ou ne correspondent pas à la configuration.
2000000	L'optimisation préalable pour le refroidissement requiert la réussite de l'optimisation préalable pour le chauffage.
4000000	Erreur au démarrage de l'optimisation fine. "Heat.EnableTuning" et "Cool.EnableTuning" ne peuvent pas être mis à 1 en même temps.
8000000	Une erreur pendant le calcul des paramètres PID a entraîné des paramètres invalides (par exemple un gain négatif, les paramètres PID actuels ne sont pas modifiés et l'optimisation n'a pas d'effet).

- 1 Note : Si l'état automatique était actif avant que l'erreur ne se produise et ActivateRecoverMode a la valeur TRUE, PID_Temp reste en mode automatique.
- 2 Note : Si l'optimisation préalable ou l'optimisation fine était active avant que l'erreur ne se produise et si ActivateRecoverMode a la valeur TRUE, PID_Temp passe à l'état de fonctionnement qui est enregistré pour le paramètre Mode.
- 3 Note : Si le mode automatique était actif avant que l'erreur ne se produise et si ActivateRecoverMode a la valeur TRUE, PID_Compact délivre la valeur de réglage de remplacement configurée. Dès que l'erreur n'est plus en attente, PID_Temp repasse en mode automatique.
- 4 Note : Si ActivateRecoverMode a la valeur TRUE avant que l'erreur ne se produise, PID_Temp annule l'optimisation et passe à l'état de fonctionnement qui est enregistré pour le paramètre Mode.

8.8 Configuration des contrôleurs PID_Compact et PID_3Step

Les paramètres de l'objet technologique déterminent le fonctionnement du régulateur PID. Utilisez l'icône pour ouvrir l'éditeur de configuration.

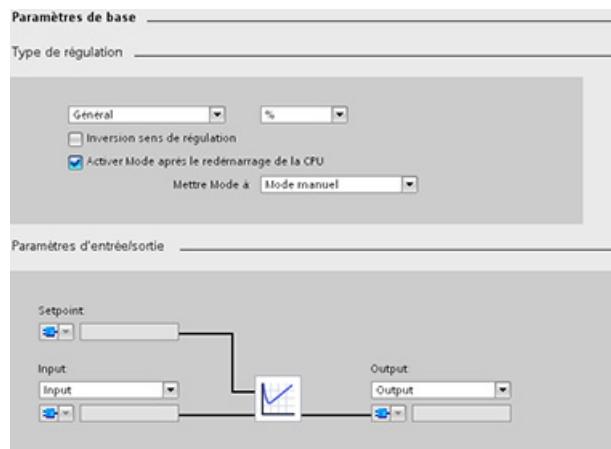


Tableau 8- 12 Exemple de paramètres de configuration pour l'instruction PID_Compact

Paramètres	Description
Paramètres de base	Type de régulation Sélectionne les unités physiques.
	Inversion du sens de régulation Permet la sélection d'une boucle PID à action inverse. <ul style="list-style-type: none"> Si la case n'est pas cochée, la boucle PID est à action directe et la valeur de réglage de la boucle PID augmente si la mesure est inférieure à la consigne. Si cette case est cochée, la valeur de réglage de la boucle PID augmente si la mesure est supérieure à la consigne.
	Activer le dernier mode après redémarrage de la CPU Redémarre la boucle PID après sa réinitialisation ou si une limite de mesure a été dépassée et est revenue dans la plage valide.
	Mesure Sélectionne soit le paramètre Input, soit le paramètre Input_PER (analogique) pour la mesure. Input_PER peut provenir directement d'un module d'entrées analogiques.
	Valeur de réglage Sélectionne soit le paramètre Output, soit le paramètre Output_PER (analogique) pour la valeur de réglage. Output_PER peut aller directement à un module de sorties analogiques.
Mesure	Met à l'échelle la plage ainsi que les limites pour la mesure. Si la mesure devient inférieure à la limite inférieure ou supérieure à la limite supérieure, la boucle PID passe en mode inactif et met la valeur de réglage à 0. Pour utiliser Input_PER, vous devez mettre à l'échelle la mesure analogique.

8.8 Configuration des contrôleurs PID_Compact et PID_3Step

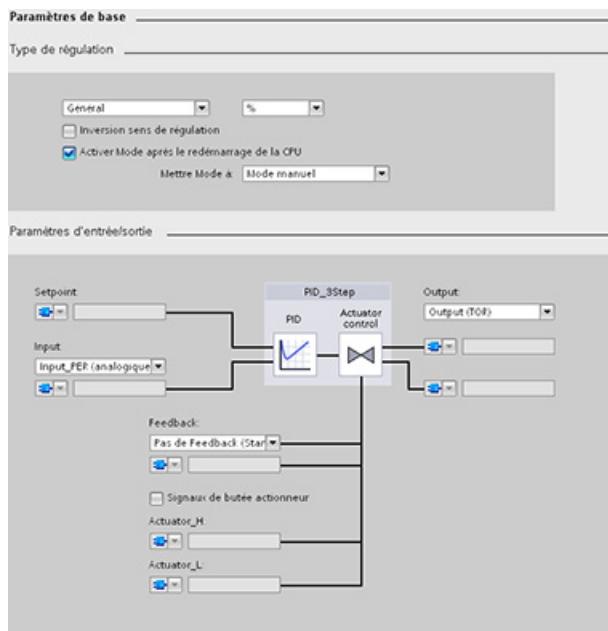


Tableau 8- 13 Exemple de paramètres de configuration pour l'instruction PID_3Step

Paramètres	Description
basiques	Type de régulateur
	Sélectionne les unités physiques.
	Inverse le sens de régulation
	Permet la sélection d'une boucle PID à action inverse. <ul style="list-style-type: none"> Si la case n'est pas cochée, la boucle PID est à action directe et la valeur de réglage de la boucle PID augmente si la mesure est inférieure à la consigne. Si cette case est cochée, la valeur de réglage de la boucle PID augmente si la mesure est supérieure à la consigne.
	Activer le mode après le démarrage de la CPU
	Redémarre la boucle PID après sa réinitialisation ou si une limite de mesure a été dépassée et est revenue dans la plage valide. Paramétrier le mode sur : Définit le mode dans lequel l'utilisateur veut que le PID aille après le redémarrage.
Entrée	Sélectionne soit le paramètre Input, soit le paramètre Input_PER (analogique) pour la mesure. Input_PER peut provenir directement d'un module d'entrées analogiques.
Sortie	Sélectionne soit l'utilisation de sorties TOR (Output_UP et Output_DN), soit l'utilisation d'une sortie analogique (Output_PER) pour la valeur de réglage.
Rétroaction	Sélectionne le type d'état renvoyé par la boucle PID : <ul style="list-style-type: none"> Pas de rétroaction (valeur par défaut) Rétroaction Feedback_PER
Mesure	Met à l'échelle la plage ainsi que les limites pour la mesure. Si la mesure devient inférieure à la limite inférieure ou supérieure à la limite supérieure, la boucle PID passe en mode inactif et met la valeur de réglage à 0. Pour utiliser Input_PER, vous devez mettre à l'échelle la mesure analogique (valeur d'entrée).

Paramètres		Description
Actionneur	Temps de transition du moteur	Définit le temps nécessaire pour passer de l'état ouvert à l'état fermé de la vanne (vous trouverez cette valeur sur la fiche technique ou la plaque signalétique de la vanne).
	Temps d'activation minimum	Définit le temps de mouvement minimum de la vanne (vous trouverez cette valeur sur la fiche technique ou la plaque signalétique de la vanne).
	Temps de désactivation minimum	Définit le temps de pause minimum de la vanne (vous trouverez cette valeur sur la fiche technique ou la plaque signalétique de la vanne).
	Réaction à l'erreur	Définit le comportement de la vanne lorsqu'une erreur est détectée ou que la boucle PID est réinitialisée. Si vous optez pour l'utilisation d'une position de substitution, entrez la "Position de sécurité". Pour une rétroaction ou une sortie analogique, sélectionnez une valeur entre la limite supérieure et la limite inférieure pour la valeur de réglage. Pour des sorties TOR, vous pouvez choisir 0% (désactivé) ou 100% (activé).
	Mise à l'échelle de la rétroaction de position ¹	<ul style="list-style-type: none"> "Butée supérieure" et "Butée inférieure" définissent respectivement la position positive maximale (entièrement ouvert) et la position négative maximale (entièrement fermé). "Butée supérieure" doit être supérieure à "Butée inférieure". "Limite supérieure de la mesure" et "Limite inférieure de la mesure" définissent les positions supérieure et inférieure de la vanne pendant le réglage et le mode automatique. "FeedbackPER" ("Low" et "High") définit la rétroaction analogique de la position de la vanne. "FeedbackPER High" doit être supérieur à "FeedbackPER Low".
Paramètres avancés	Surveillance de la mesure	Définit les limites d'avertissement inférieure et supérieure pour la mesure.
	Paramètres PID	Si l'utilisateur le souhaite, il peut entrer ses propres paramètres d'optimisation PID dans cette fenêtre. La case "Activer l'entrée manuelle" doit être cochée pour permettre cela.

¹ "Mise à l'échelle de la rétroaction de position" ne peut être modifié que si vous avez activé "Rétroaction" dans les paramètres de base.

8.9 Configuration du régulateur PID_Temp

Les paramètres de l'objet technologique déterminent le fonctionnement du régulateur PID. Utilisez l'icône pour ouvrir l'éditeur de configuration.

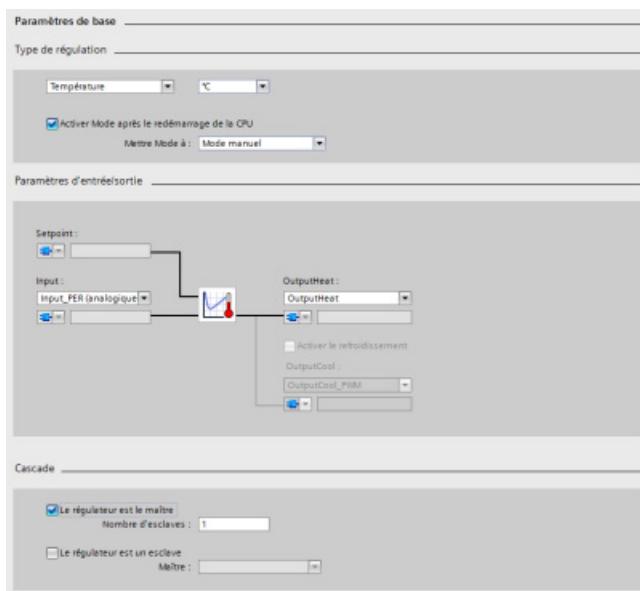


Tableau 8- 14 Exemple de paramètres de configuration pour l'instruction PID_Temp

Paramètres		Description
basiques	Type de régulateur	Sélectionne les unités physiques.
	Activer le mode après le démarrage de la CPU	Redémarre la boucle PID après sa réinitialisation ou si une limite de mesure a été dépassée et est revenue dans la plage valide. Paramétrer le mode sur : Définit le mode dans lequel l'utilisateur veut que le PID aille après le redémarrage.
	Entrée	Sélectionne soit le paramètre Input, soit le paramètre Input_PER (analogique) pour la mesure. Input_PER peut provenir directement d'un module d'entrées analogiques.
	Sortie chauffage	Sélectionne soit l'utilisation de sorties TOR (OutputHeat et OutputHeat_PWM), soit l'utilisation d'une sortie analogique (OutputHeat_PER (analog)) pour la valeur de réglage.
	Sortie Refroidissement	Sélectionne soit l'utilisation de sorties TOR (OutputCool et OutputCool_PWM), soit l'utilisation d'une sortie analogique (OutputCool_PER (analog)) pour la valeur de réglage.

Paramètres		Description	
Mesure	<p>Met à l'échelle la plage ainsi que les limites pour la mesure. Si la mesure devient inférieure à la limite inférieure ou supérieure à la limite supérieure, la boucle PID passe en mode inactif et met la valeur de réglage à 0.</p> <p>Pour utiliser Input_PER, vous devez mettre à l'échelle la mesure analogique (valeur d'entrée).</p>		
Cascade	Le régulateur est un maître	Définit le régulateur comme maître et sélectionne le nombre d'esclaves.	
	Le régulateur est un esclave.	Définit le régulateur comme esclave et sélectionne le nombre de maîtres.	

Type de régulateur

Para-mètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Quantité physique	"PhysicalQuantity"	En-tier (énum)	<ul style="list-style-type: none"> • Général • Température (=par défaut) 	<p>Pré-sélection de la valeur de l'unité physique</p> <p>Pas de commande multi-valeur et pas de possibilité d'édition dans le mode en ligne de la vue fonctionnelle.</p>
Unité de mesure	"PhysicalUnit"	En-tier (énum)	<ul style="list-style-type: none"> • Général : Unités = % • Température : Unités (sélections possibles) = <ul style="list-style-type: none"> – °C (= par défaut) – °F – K 	<p>La sélection personnalisée de l'unité est remise à "0" si vous modifiez la quantité physique.</p>

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Activer le mode après le démarrage de la CPU	"RunModeByStartup"	Booléen	Case à cocher	Si TRUE est activé (= réglage par défaut), le régulateur passe à l'état enregistré dans la variable "Mode" après un cycle d'alimentation (alimentation allumée, éteinte puis rallumée) ou une commutation ARRÊT-DEMARRAGE de l'API. En cas contraire, PID_Temp reste en mode "inactif".
Paramétrer le mode sur	"Mode"	Entier (énum)	Modes (sélections possibles) : <ul style="list-style-type: none"> • 0: Inactif • 1: Optimisation préalable • 2: Optimisation fine • 3: Mode automatique • 4: Mode manuel (=valeur par défaut) 	La station d'ingénierie (ES) définit la valeur de démarrage de la variable "Mode" selon la sélection de l'utilisateur. La valeur par défaut de "Mode" (enregistrée dans TO-DB) est Mode manuel.

Paramètres d'entrée/de sortie

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Consigne	Consigne	(Nombre réel)	Nombre réel	Accessible uniquement sur la page Propriétés. Pas de commande multi-valeur en mode en ligne de la vue fonctionnelle.
Sélection de l'entrée	"Config.InputPerOn"	Booléen (énum)	Booléen	Sélectionne le type d'entrée à utiliser. Sélections possibles : <ul style="list-style-type: none"> • FALSE : "Entrée" (nombre réel) • TRUE : "Input_PER (analogique)"

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Entrée	Input ou Input_PER	Réel ou entier	Réel ou entier	Accessible uniquement sur la page Propriétés. Pas de commande multi-valeur en mode en ligne de la vue fonctionnelle.
Sélection de la sortie (chauffage)	"Config.Output.Heat.Select"	Entier (énum)	2 >= Config.Output.Heat.Select >= 0	Sélectionne le type de sortie à utiliser pour le chauffage. Sélections possibles : <ul style="list-style-type: none">• "OutputHeat" (nombre réel)• "OutputHeat_PWM" (Booléen) (=par défaut)• "OutputHeat_PER (analogique)" (mot) Est réglé une fois sur "OutputHeat" si la case "Ce régulateur est un maître" est cochée par l'utilisateur dans la partie "Cascade".
Sortie (chauffage)	OutputHeat, OutputHeat_PER, ou OutputHeat_PWM	Réel ou entier ou booléen	Réel, entier ou booléen	Accessible uniquement sur la page Propriétés. Pas de commande multi-valeur en mode en ligne de la vue fonctionnelle.

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Activer sortie (refroidissement)	"Config.ActivateCooling"	Booléen	Booléen	<p>Cocher cette case :</p> <ul style="list-style-type: none">• règle "Config.OutputHeat.PidLowerLimit sur 0,0 une fois.• règle le paramètre "Config.ActivateCooling" sur TRUE, au lieu de FALSE si la case n'est pas cochée (=valeur par défaut).• Active toutes les autres commandes "Sortie (refroidissement)" (dans "Paramètres de base" et d'autres vues).• Fait passer la ligne allant du symbole PID aux commandes de gris à noir.• La case "Ce régulateur est un maître" est décochée dans la partie Cascade. <p>Remarque : Disponible seulement si vous ne configurez pas le régulateur comme un maître d'une cascade (la case "Ce régulateur est un maître" est décochée dans la partie "Cascade" ; "Config.Cascade.IsMaster" = FALSE).</p>

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Sélection de la sortie (refroidissement)	"Config.Output.Cool.Select"	Entier (énum)	2 >= Config.Output.Heat.Select >= 0	<p>Sélectionne le type de sortie à utiliser pour le refroidissement.</p> <p>Sélections possibles :</p> <ul style="list-style-type: none"> • "OutputCool" (nombre réel) • "OutputCool_PWM" (Booléen) (=par défaut) • "OutputCool_PER (analogique)" (mot) <p>Disponible seulement si vous cochez "Activer sortie (refroidissement)" ; (Config.ActivateCooling = TRUE).</p>
Sortie (refroidissement)	OutputCool, OutputCool_PER, ou OutputCool_PWM	Réel ou entier ou booléen	Réel, entier ou booléen	Accessible uniquement sur la page Propriétés. Pas de commande multivaleur en mode en ligne de la vue fonctionnelle.

Paramètres de la cascade

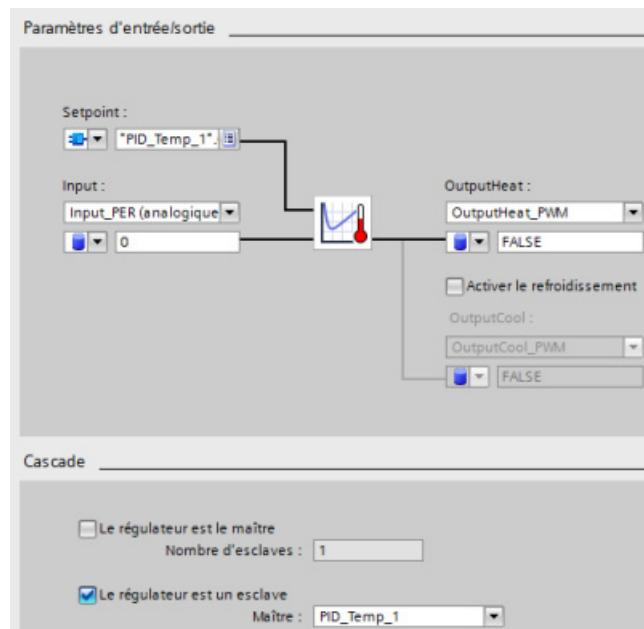
Les paramètres suivants vous permettent de sélectionner des régulateurs comme maîtres ou esclaves et de déterminer le nombre de régulateurs esclaves recevant leur consigne directement du régulateur maître :

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Ce régulateur est un maître	"Config.Cascade.IsMaster"	Booléen	Booléen	<p>Montre si ce régulateur est un maître dans une cascade. Lorsque vous cochez cette case, vous effectuez ce qui suit :</p> <ul style="list-style-type: none">• régler le paramètre "Config.Cascade.IsMaster" sur TRUE, au lieu de FALSE si la case n'est pas cochée (=valeur par défaut).• régler une fois "Sélection de la sortie (chauffage)" sur "OutputHeat" dans la partie "Paramètres d'entrée/sortie" (Config.Output.Heat.Select = 0).• activer le champ de saisie "Nombre d'esclaves".• décocher la case "Activer sortie (refroidissement)" dans la partie "Paramètres d'entrée/sortie". <p>Remarque : Disponible seulement si la sortie de refroidissement de ce régulateur est désactivée (la case "Activer sortie (refroidissement)" dans la partie "Paramètres d'entrée/sortie" est décochée (Config.ActivateCooling = FALSE)).</p>

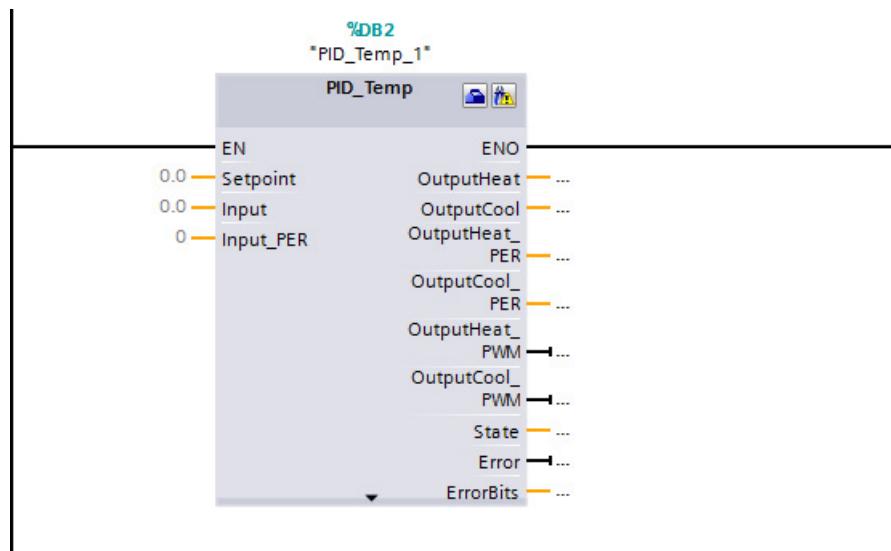
Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Nombre d'es-claves	"Config.Cascade.CountSlaves"	Ent	255 >= Config.Cascade.CountSlaves >= 1	Nombre de régulateurs esclaves recevant leur consigne directement de ce régulateur maître. L'instruction PID_Temp traite cette valeur, ainsi que d'autres, pour la gestion de l'emballage. "Nombre d'es-claves" n'est disponible que si la case "Ce régulateur est un maître" est activée (Config.Cascade.IsMaster = TRUE).
Ce régulateur est un esclave	"Config.Cascade.IsSlave"	Booléen	Booléen	Montre si ce régulateur est un esclave dans une cascade. Lorsque vous cochez cette case, vous réglez le paramètre "Config.Cascade.IsSlave" sur TRUE, au lieu de FALSE si la case n'est pas cochée (=valeur par défaut). Vous devez cocher cette case dans la page des propriétés pour activer la liste déroulante "SelectionMaster".

Exemple : Régulateurs en cascade

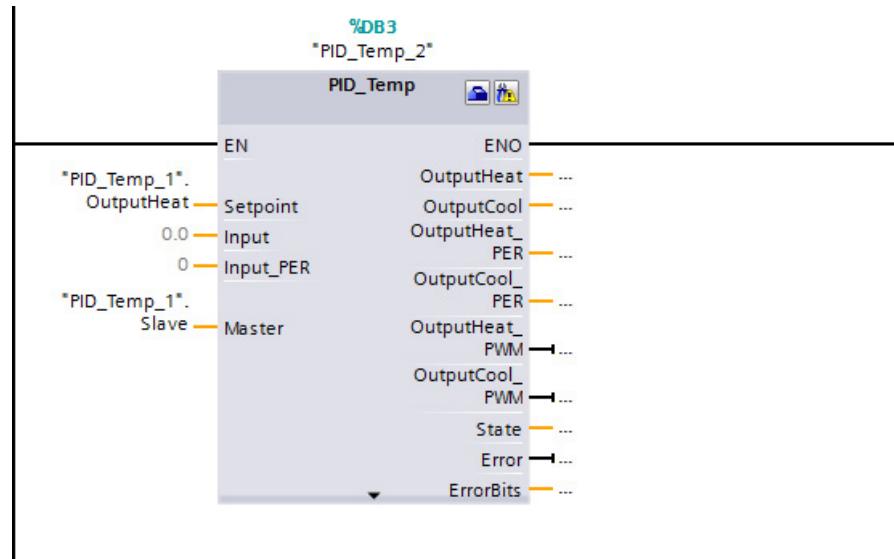
Dans la boîte de dialogue "Paramètres de base" ci-dessous, vous voyez la partie "Paramètres d'entrée/sortie" et la partie "Cascade" pour le régulateur esclave "PID_Temp_2" après avoir sélectionné "PID_Temp_1" comme maître. Vous établissez la connexion entre le régulateur maître et le régulateur esclave :



Réseau 1 : Dans ces réseaux, vous établissez la connexion entre le maître "PID_Temp_1" et l'esclave "PID_Temp_2" dans l'éditeur de programmation :



Réseau 2 : Vous établissez la connexion entre les paramètres "OutputHeat" et "Slave" du maître "PID_Temp_1" et les paramètres "Setpoint" et "Master" de l'esclave "PID_Temp_2", respectivement :



Auto-optimisation de processus thermiques

L'instruction PID_Temp propose deux modes pour l'auto-optimisation :

- "Optimisation préalable" (paramètre "Mode" = 1)
- "Optimisation fine" (paramètre "Mode" = 2)

Selon la configuration du régulateur, différentes variantes de ces méthodes d'optimisation sont disponibles :

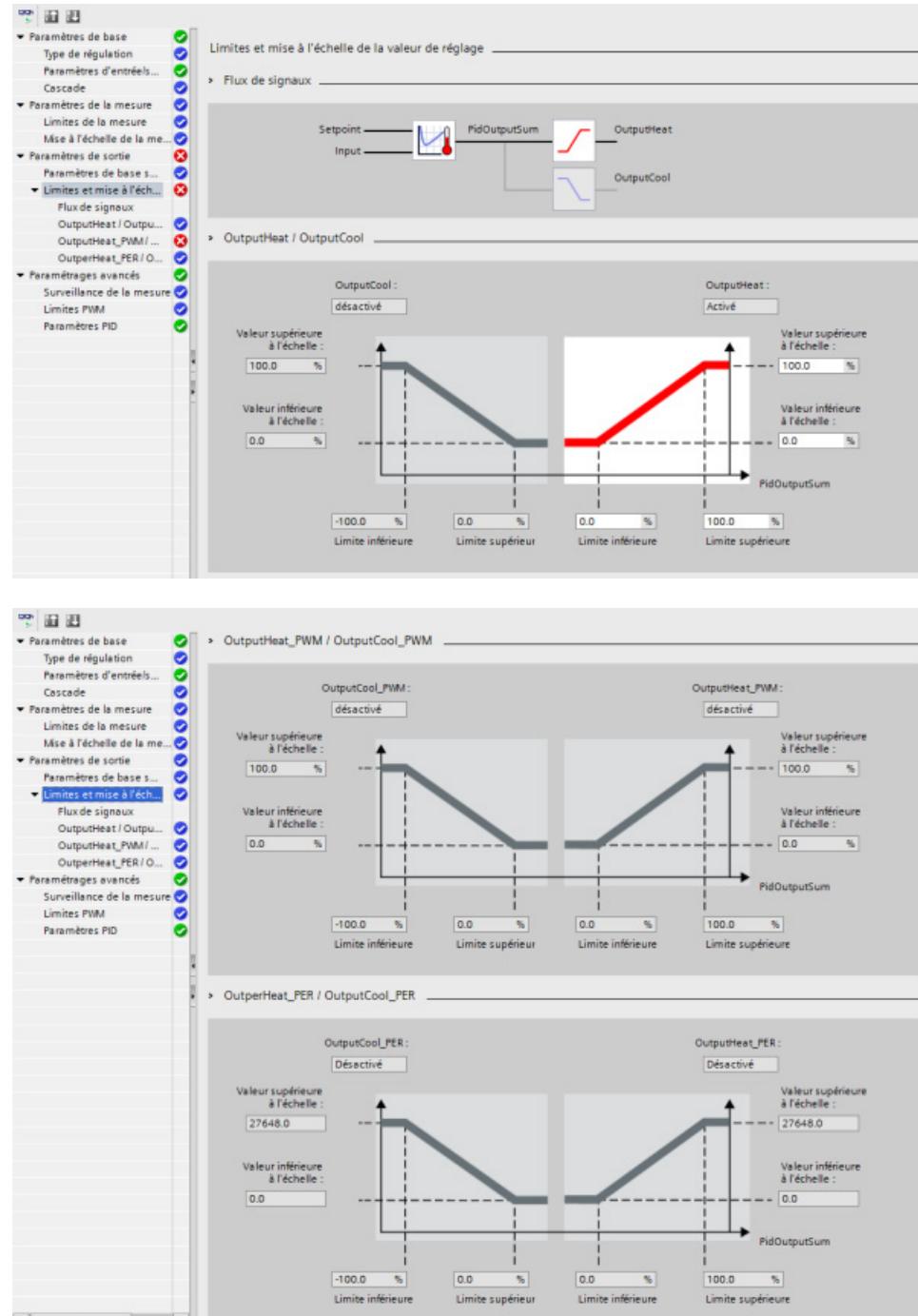
Configuration	Régulateur avec sortie de chauffage	Régulateur avec sortie de chauffage et de refroidissement utilisant un facteur de refroidissement	Régulateur avec sortie de chauffage et de refroidissement utilisant deux jeux de paramètres PID
Valeurs TO-DB associées	<ul style="list-style-type: none">Config.ActivateCooling = FALSEConfig.AdvancedCooling = non pertinent	<ul style="list-style-type: none">Config.ActivateCooling = TRUEConfig.AdvancedCooling = FALSE	<ul style="list-style-type: none">Config.ActivateCooling = TRUEConfig.AdvancedCooling = TRUE
Méthodes d'optimisation disponibles	<ul style="list-style-type: none">"Optimisation préalable du chauffage""Optimisation fine du chauffage" (le décalage de refroidissement ne peut pas être utilisé)	<ul style="list-style-type: none">"Optimisation préalable du chauffage""Optimisation fine du chauffage" (le décalage de refroidissement peut être utilisé)	<ul style="list-style-type: none">"Optimisation préalable du chauffage et du refroidissement""Optimisation préalable du chauffage""Optimisation préalable du refroidissement""Optimisation fine du chauffage" (le décalage de refroidissement peut être utilisé)"Optimisation fine du refroidissement" (le décalage de chauffage peut être utilisé)

Limites et mise à l'échelle des valeurs de réglage

Activation du refroidissement désactivée

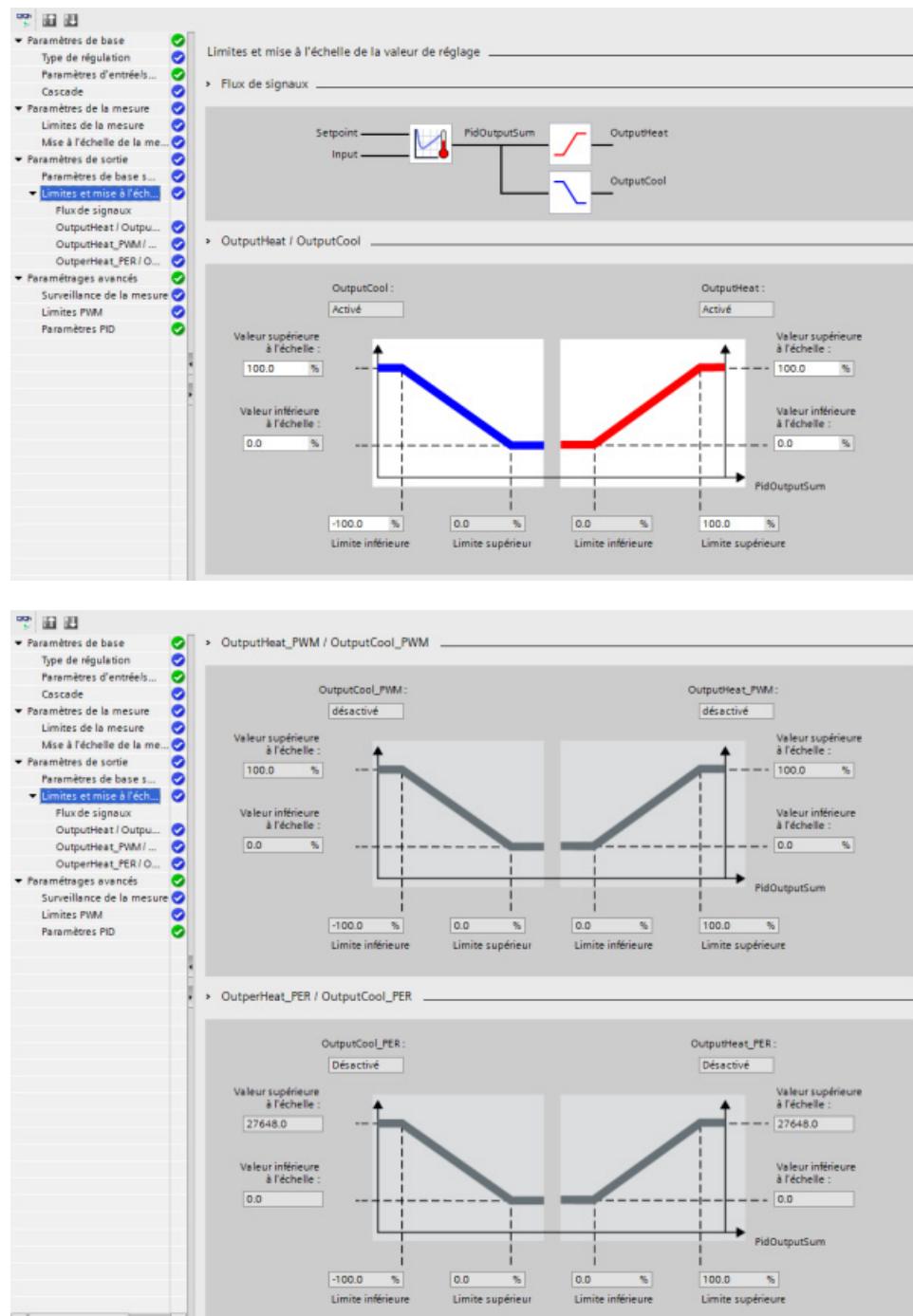
Si vous configurez l'instruction PID_Temp comme maître pour une cascade, la case "Activer sortie (refroidissement)" est décochée et désactivée dans la vue "Paramètres de base" et tous les paramètres dans la vue "Paramètres de sortie" dépendant de l'activation du refroidissement sont également désactivés.

La figure ci-dessous représente la partie "Limites et mise à l'échelle de la valeur de réglage" dans la vue "Paramètres de sortie" avec refroidissement désactivé (OutputHeat_PWM sélectionné dans la vue "Paramètres d'entrée/sortie" et OutputHeat toujours activé) :



Activation du refroidissement activé

La figure ci-dessous représente la partie "Limites et mise à l'échelle de la valeur de réglage" dans la vue "Paramètres de sortie" avec refroidissement activé (OutputCool_PER et OutputHeat_PWM sélectionné dans la vue "Paramètres d'entrée/sortie" ; OutputCool et OutputHeat toujours activés) :



Etats de fonctionnement

Pour modifier l'état de fonctionnement manuellement, l'utilisateur doit régler le paramètre d'entrée-sortie "Mode" du régulateur et l'activer en faisant passer "ModeActivate" de FALSE à TRUE (front montant déclenché). Vous devez réinitialiser "ModeActivate" avant le prochain changement de mode ; il n'est pas automatiquement réinitialisé.

Le paramètre de sortie "State" montre l'état de fonctionnement actuel et est mis sur le "Mode" requis, si possible. Le paramètre "State" ne peut pas être modifié directement ; il ne peut être modifié que par l'intermédiaire du paramètre "Mode" ou par des changements du mode automatique par le régulateur.

"Mode" / "State"	Nom	Description
0	Inactif	<p>L'instruction PID_Temp :</p> <ul style="list-style-type: none"> désactive l'algorithme PID et la modulation de largeur d'impulsion Met à "0" (FALSE) toutes les sorties du régulateur (OutputHeat, OutputCool, OutputHeat_PWM, OutputCool_PWM, OutputHeat_PER, OutputCool_PER), quels que soient les limites de sortie ou les décalages configurés. Vous pouvez atteindre ce mode en paramétrant "Mode" = 0, "Reset" = TRUE, ou par erreur.
1	Optimisation préalable (optimisation au démarrage / SUT)	<p>Ce mode détermine les paramètres pendant le premier démarrage du régulateur. Contrairement à PID_Compact, pour PID_Temp, vous devez indiquer si vous avez besoin d'une optimisation de chauffage, d'une optimisation de refroidissement ou des deux à l'aide des paramètres "Heat.EnableTuning" et "Cool.EnableTuning".</p> <p>Vous pouvez activer "Pretuning" (optimisation préalable) à partir du Mode inactif, Mode automatique ou Mode manuel.</p> <p>Si l'optimisation est réussie, PID_Temp passe en mode automatique. Si l'optimisation n'a pas réussi, la commutation de l'état de fonctionnement dépend de "ActivateRecoverMode".</p>
2	Optimisation fine (optimisation en cours de fonctionnement / TIR)	<p>Ce mode détermine le paramétrage optimal du régulateur PID à la consigne. Contrairement à PID_Compact, pour PID_Temp, vous devez indiquer si vous avez besoin d'une optimisation de chauffage ou d'une optimisation de refroidissement à l'aide des paramètres "Heat.EnableTuning" et "Cool.EnableTuning".</p> <p>Vous pouvez activer "Finetuning" (optimisation fine) à partir du Mode inactif, Mode automatique ou Mode manuel.</p> <p>Si l'optimisation est réussie, PID_Temp passe en mode automatique. Si l'optimisation n'a pas réussi, la commutation de l'état de fonctionnement dépend de "ActivateRecoverMode".</p>
3	Mode automatique	<p>En mode automatique (le mode de régulation PID standard), le résultat de l'algorithme PID détermine les valeurs de réglage.</p> <p>PID_Temp bascule sur Inactif si une erreur survient et que "ActivateRecoverMode" = FALSE. Si une erreur survient et que "ActivateRecoverMode" = TRUE, la commutation de l'état de fonctionnement dépend de l'erreur. Pour plus d'informations, veuillez vous référer à l'instruction PID_Temp ErrorBit parameters (Page 229).</p>

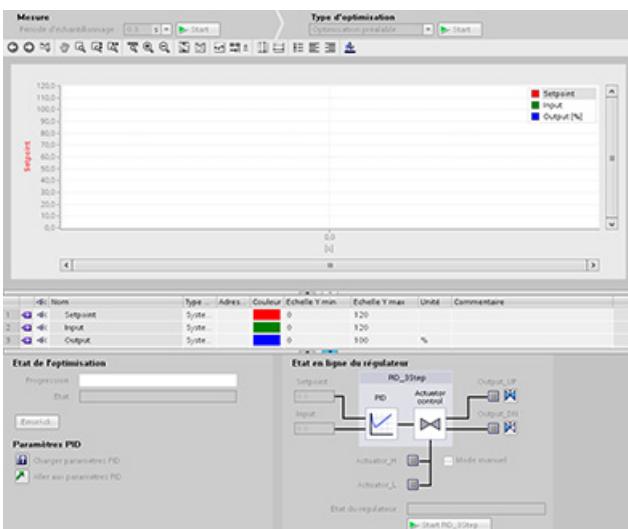
"Mode" / "State"	Nom	Description
4	Mode manuel	<p>Dans ce mode, le régulateur PID met à l'échelle, limite et transmet la valeur du paramètre "ManualValue" aux sorties.</p> <p>Le régulateur PID affecte "ManualValue" lors de la mise à l'échelle de l'algorithme PID (comme par exemple "PidOutputSum"), sa valeur décide donc si elle s'applique aux sorties de chauffage ou de refroidissement.</p> <p>Vous pouvez atteindre ce mode en paramétrant "Mode" = 4 ou ManualEnable"= TRUE.</p>
5	Valeur de réglage de remplacement avec surveillance d'erreur (mode Recover)	<p>Vous pouvez activer ce mode en réglant "Mode" sur 5. Ce mode est une réaction automatique du régulateur aux erreurs si le mode automatique est activé au moment où l'erreur se produit :</p> <ul style="list-style-type: none"> • SetSubstituteOutput = FALSE (dernière valeur de réglage valide) • SetSubstituteOutput = TRUE (valeur enregistrée dans le paramètre "SubstituteOutput") <p>Lorsque PID_Temp est en "Mode automatique" et que le paramètre "ActivateRecoverMode" = TRUE, PID_Temp passe dans ce mode si les erreurs suivantes surviennent :</p> <ul style="list-style-type: none"> • "Valeur invalide au paramètre "Input_PER". Vérifier la présence d'une erreur à l'entrée analogique (par exemple, un fil coupé)". (ErrorBits = DW#16#0002) • "Valeur invalide au paramètre "Input". La valeur n'est pas un nombre." (ErrorBits = DW#16#0200) • "Echec du calcul de la valeur de réglage. Vérifiez les paramètres PID." (ErrorBits = DW#16#0400) • "Valeur invalide au paramètre "Setpoint". La valeur n'est pas un nombre." (ErrorBits = DW#16#1000) <p>Dès que l'erreur n'est plus en attente, PID_Temp repasse automatiquement en mode automatique.</p>

8.10 Mise en service des régulateurs PID_Compact et PID_3Step

Utilisez l'éditeur de mise en service pour configurer le régulateur PID afin qu'une optimisation préalable soit effectuée à la mise en route et en fonctionnement. Pour ouvrir l'éditeur de mise en service, cliquez sur l'icône correspondante dans l'instruction ou dans le navigateur du projet.



Tableau 8- 15 Exemple d'écran de mise en service (PID_3Step)



- Mesure : Pour afficher la consigne, la mesure et la valeur de réglage dans un affichage de courbes en temps réel, entrez la période d'échantillonnage et cliquez sur le bouton "Démarrer".
- Optimisation : Pour optimiser la boucle PID, sélectionnez soit "Optimisation préalable" soit "Optimisation fine" (manuelle) et cliquez sur le bouton "Démarrer". Le régulateur PID exécute plusieurs phases pour calculer les temps de réponse et d'actualisation du système. Les paramètres optimisés appropriés sont calculés à partir de ces valeurs.

Une fois le processus d'optimisation achevé, vous pouvez stocker les nouveaux paramètres en cliquant sur le bouton "Charger les paramètres PID dans le projet" dans la partie "Paramètres PID" de l'éditeur de mise en service.

Si une erreur se produit pendant l'optimisation, la valeur de réglage de la boucle PID est mise à 0. Le régulateur PID passe alors en mode "inactif". L'état signale l'erreur.

Régulation de la valeur de démarrage PID

Vous pouvez éditer les valeurs réelles des paramètres de configuration PID afin que le comportement du régulateur PID puisse être optimisé en mode en ligne.

Ouvrez les "Objets technologiques" pour votre régulateur PID et son objet "Configuration". Pour accéder à la commande de valeur de début, cliquez sur l'"icône lunettes" dans le coin supérieur gauche de la boîte de dialogue :



Vous pouvez maintenant modifier la valeur des paramètres de configuration de votre régulateur PID tel qu'indiqué dans la figure ci-dessous.

8.10 Mise en service des régulateurs PID_Compact et PID_3Step

Vous pouvez comparer la valeur réelle à la valeur de démarrage (hors ligne) du projet et à la valeur de démarrage (en ligne) API de chaque paramètre. Ceci est nécessaire pour comparer les différences en ligne/hors ligne du bloc de données Objet technologique (TO-DB) et pour être informé des valeurs qui seront utilisées comme valeurs actuelles à la prochaine commutation Arrêt-Démarrage de l'API. De plus, une icône de comparaison donne une indication visuelle pour vous aider à identifier facilement les différences en ligne/hors ligne :



La figure ci-dessus montre l'écran de paramètre PID avec des icônes de comparaison montrant quelles valeurs sont différentes entre les projets en ligne et hors ligne. Une icône verte indique que les valeurs sont les mêmes ; une icône bleue/orange indique que les valeurs sont différentes.

En outre, cliquez sur le bouton paramètre avec la flèche vers le bas pour ouvrir une petite fenêtre qui montre la valeur de démarrage (hors ligne) du projet et sur la valeur de démarrage (en ligne) API de chaque paramètre :



8.11 Mise en service du régulateur PID_Temp

Utilisez l'éditeur de mise en service pour configurer le régulateur PID afin qu'une optimisation préalable soit effectuée à la mise en route et en fonctionnement. Pour ouvrir l'éditeur de mise en service, cliquez sur l'icône correspondante dans l'instruction ou dans le navigateur du projet.



Tableau 8- 16 Exemple d'écran de mise en service (PID_Temp)

Mesure : Pour afficher la consigne, la mesure et la valeur de réglage dans un affichage de courbes en temps réel, entrez la période d'échantillonnage et cliquez sur le bouton "Démarrer".

Optimisation : Pour optimiser la boucle PID_Temp, sélectionnez soit "Optimisation préalable" soit "Optimisation fine" (manuelle) et cliquez sur le bouton "Démarrer". Le régulateur PID exécute plusieurs phases pour calculer les temps de réponse et d'actualisation du système. Les paramètres optimisés appropriés sont calculés à partir de ces valeurs.

Une fois le processus d'optimisation achevé, vous pouvez enregistrer les nouveaux paramètres en cliquant sur le bouton "Charger les paramètres PID dans le projet" dans la partie "Paramètres PID" de l'éditeur de mise en service.

Si une erreur se produit pendant l'optimisation, la valeur de réglage de PID se met à "0". Le mode de PID est ensuite réglé sur le mode "inactif". L'état signale l'erreur.

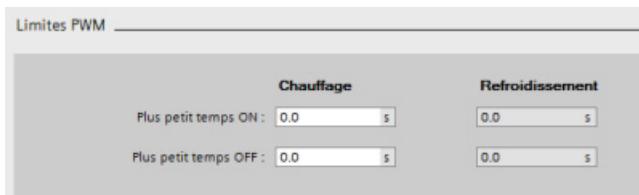
Limites PWM

Les actionneurs commandés par la fonction logicielle PWM de PID_Temp peuvent nécessiter une protection contre les durées d'impulsion trop courtes (par exemple un relais à thyristor doit être activé pendant plus de 20 ms avant de pouvoir même réagir) ; vous affectez un temps d'activation minimum. L'actionneur peut également négliger les impulsions courtes et donc corrompre la qualité de la régulation. Un temps de désactivation minimum peut s'avérer nécessaire (par exemple pour éviter la surchauffe).

Pour afficher la vue des limites PWM, vous devez ouvrir la vue fonctionnelle dans la configuration des objets technologiques (TO) et sélectionner "Limites PWM" dans le nœud "Paramètres avancées" dans l'arborescence de navigation.

Si vous ouvrez la vue "Limites PWM" dans la vue fonctionnelle et que vous activez la surveillance "bouton "lunettes""), tous les contrôles indiquent la valeur de surveillance en ligne de TO-DB avec un arrière-plan de couleur orange et le contrôle multi-valeur, et vous pouvez éditer les valeurs (si les conditions de configuration sont remplies ; voir le tableau ci-dessous).

8.11 Mise en service du régulateur PID_Temp



Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Temps d'activation minimum temps (chauffage) 1,2	"Config.Output.Heat.MinimumOnTime"	Nombre réel	100000.0 >= "Config.Output.Heat.MinimumOnTime >= 0.0	Une impulsion à OutputHeat_PWM n'est jamais plus courte que cette valeur.
Temps de désactivation minimum temps (chauffage) 1,2	"Config.Output.Heat.MinimumOffTime"	Nombre réel	100000.0 >= "Config.Output.Heat.MinimumOffTime >= 0.0	Une rupture à OutputHeat_PWM n'est jamais plus courte que cette valeur.
Temps d'activation minimum temps (refroidissement) 1,3,4	"Config.Output.Cool.MinimumOnTime"	Nombre réel	100000.0 >= Config.Output.Cool.MinimumOnTime >= 0.0	Une impulsion à OutputCool_PWM n'est jamais plus courte que cette valeur.
Temps de désactivation minimum temps (refroidissement) 1,3,4	"Config.Output.Cool.MinimumOffTime"	Nombre réel	100000.0 >= Config.Output.Cool.MinimumOffTime >= 0.0	Une rupture à OutputCool_PWM n'est jamais plus courte que cette valeur.

- 1 Ce champ affiche "s" (secondes) comme unité de temps.
- 2 Si la sélection de la Sortie (chauffage) dans la vue "Paramètres de base" n'est pas "OutputHeat_PWM" (Config.Output.Heat.Select = TRUE), vous devez régler cette valeur sur "0,0".
- 3 Si la sélection de la Sortie (refroidissement) dans la vue "Paramètres de base" n'est pas "OutputCool_PWM" (Config.Output.Cool.Select = TRUE), vous devez régler cette valeur sur "0,0".
- 4 Disponible seulement si vous cochez "Activer sortie (refroidissement)" dans "Paramètres de base" (Config.ActivateCooling = TRUE).

Paramètres PID

La partie "Paramètres PID" de la vue "Paramètres avancés" est affichée en-dessous avec la fonction refroidissement et/ou "Commutation des paramètres PID" désactivées.



Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Activer l'entrée manuelle	"Retain.CtrlParams. SetByUser"	Booléen		Vous devez cocher cette case pour saisir les paramètres PID manuellement.
Gain proportionnel (chauffage) 2	"Retain.CtrlParams. Heat.Gain"	Nombre réel	Gain $\geq 0,0$	Gain proportionnel PID pour le chauffage
Temps d'intégration (chauffage) 1,2	"Retain.CtrlParams. Heat.Ti"	Nombre réel	100000.0 \geq Ti $\geq 0,0$	Action intégrale PID pour le chauffage.
Temps de dérivation (chauffage) 1,2	"Retain.CtrlParams. Heat.Td"	Nombre réel	100000.0 \geq Td $\geq 0,0$	Temps de dérivation PID pour le chauffage.
Coefficient de retard de l'action dérivée (chauffage) 2	"Retain.CtrlParams. Heat.TdFiltRatio"	Nombre réel	TdFiltRatio $\geq 0,0$	Coefficient de retard de l'action dérivée PID pour le chauffage qui définit le temps de retard de l'action dérivée sous forme de coefficient du temps de dérivation PID.
Pondération de l'action proportionnelle(chauffage) 2	"Retain.CtrlParams. Heat.PWeighting"	Nombre réel	1,0 \geq PWeighting $\geq 0,0$	Pondération du gain proportionnel PID pour le chauffage soit dans le circuit de régulation directe soit dans la boucle de régulation.

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Pondération de l'action dérivée (chauffage) ²	"Retain.CtrlParams.Heat.DWeighting"	Nombre réel	$1,0 \geq DWeighting \geq 0,0$	Pondération de la partie de dérivation PID pour le chauffage soit dans le circuit de régulation directe soit dans la boucle de régulation.
Période d'échantillonnage de l'algorithme PID (chauffage) ^{1,2}	"Retain.CtrlParams.Heat.Cycle"	Nombre réel	$100000,0 \geq Cycle > 0,0$	Cycle d'appel interne du régulateur PID pour le chauffage. Arrondi à un multiple entier du temps de cycle d'appel du FB.
Largeur de bande morte(chauffage) ^{2,3}	"Retain.CtrlParams.Heat.DeadZone"	Nombre réel	$Zone morte \geq 0,0$	Largeur de la bande morte pour l'écart de régulation de chauffage.
Zone de régulation (chauffage) ^{2,3}	"Retain.CtrlParams.Heat.ControlZone"	Nombre réel	$Zone de régulation > 0,0$	Largeur de la zone d'écart de régulation pour le chauffage lorsque la régulation PID est active. Si l'écart de régulation quitte cette plage, la sortie est commutée sur des valeurs de réglage maximum. La valeur par défaut est "MaxReal" ainsi la zone de régulation est désactivée tant que l'auto-optimisation n'est pas exécutée. La valeur "0,0" est interdite pour la Zone de régulation ; avec la valeur "0,0", PID_Temp se comporte comme un régulateur tout ou rien qui chauffe ou refroidit toujours à pleine puissance.

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Structure du régulateur (chauffage)	"PIDSelfTune.SUT.TuneRuleHeat", "PIDSelfTune.TIR.TuneRuleHeat"	Ent	"PID-SelfTune.SUT.TuneRuleHeat" = 0..2, "PID-SelfTune.TIR.TuneRuleHeat" = 0..5	<p>Vous pouvez sélectionner l'algorithme d'optimisation pour le chauffage.</p> <p>Sélections possibles :</p> <ul style="list-style-type: none"> • PID (Température) (=par défaut) ("PIDSelfTune.SUT.TuneRuleHeat" = 2) ("PIDSelfTune.TIR.TuneRuleHeat" = 0) • PI ("PIDSelfTune.SUT.TuneRuleHeat" = 0) ("PIDSelfTune.TIR.TuneRuleHeat" = 0) • PI ("PIDSelfTune.SUT.TuneRuleHeat" = 1) ("PIDSelfTune.TIR.TuneRuleHeat" = 4) <p>Toute autre combinaison affiche "Personnalisé" mais "Personnalisé" n'est pas réglé par défaut. "PID (Température)" est nouveau pour PID_Temp, avec une méthode de pré-optimisation (SUT) spécifique pour les processus de température.</p>
Gain proportionnel (refroidissement) ⁴	"Retain.CtrlParams.Cool.Gain"	Nombre réel	Gain >= 0,0	Gain proportionnel PID pour le refroidissement
Temps d'intégration (refroidissement) ^{1,4}	"Retain.CtrlParams.Cool.Ti"	Nombre réel	100000,0 >= Ti >= 0,0	Action intégrale PID pour le refroidissement

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Temps de dérivation (refroidissement) ^{1,4}	"Retain.CtrlParams.Cool.Td"	Nombre réel	100000,0 >= Td >= 0,0	Temps de dérivation PID pour le refroidissement.
Coefficient de retard de l'action dérivée (refroidissement) ⁴	Retain.CtrlParams.Cool.TdFiltRatio"	Nombre réel	TdFiltRatio >= 0,0	Coefficient de retard de l'action dérivée PID pour le refroidissement qui définit le temps de retard de l'action dérivée sous forme de coefficient du temps de dérivation PID.
Pondération de l'action proportionnelle (refroidissement) ⁴	"Retain.CtrlParams.Cool.PWeighting"	Nombre réel	1,0 >= PWeighting >= 0,0	Pondération du gain proportionnel PID pour le refroidissement soit dans le circuit de régulation directe soit dans la boucle de régulation.
Pondération de l'action dérivée (refroidissement) ⁴	Retain.CtrlParams.Cool.DWeighting"	Nombre réel	1,0 >= DWeighting >= 0,0	Pondération de la partie de dérivation PID pour le refroidissement soit dans le circuit de régulation directe soit dans la boucle de régulation.
Période d'échantillonnage de l'algorithme PID (refroidissement) ^{1,4}	"Retain.CtrlParams.Cool.Cycle"	Nombre réel	100000,0 >= Cycle > 0,0	Cycle d'appel interne du régulateur PID pour le refroidissement. Arrondi à un multiple entier du temps de cycle d'appel du FB.
Largeur de bande morte (refroidissement) ^{3,4}	"Retain.CtrlParams.Cool.DeadZone"	Nombre réel	Zone morte >= 0,0	Largeur de la bande morte pour l'écart de régulation de refroidissement.

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Zone de régulation (refroidissement) ^{3,4}	"Retain.CtrlParams.Cool.ControlZone"	Nombre réel	Zone de régulation > 0,0	<p>Largeur de la zone d'écart de régulation pour le refroidissement lorsque la régulation PID est active. Si l'écart de régulation quitte cette plage, la sortie est commutée sur des valeurs de réglage maximum.</p> <p>La valeur par défaut est "MaxReal" ainsi la zone de régulation est désactivée tant que l'auto-optimisation n'est pas exécutée.</p> <p>La valeur "0,0" est interdite pour la Zone de régulation ; avec la valeur "0,0", PID_Temp se comporte comme un régulateur tout ou rien qui chauffe ou refroidit toujours à pleine puissance.</p>

Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description
Structure du régulateur (refroidissement)	"PIDSelfTune.SUT.TuneRuleCool", "PIDSelfTune.TIR.TuneRuleCool"	Ent	"PID-SelfTune.SUT.TuneRuleHeat" = 0..2, "PID-SelfTune.TIR.TuneRuleHeat" = 0..5	<p>Vous pouvez sélectionner l'algorithme d'optimisation pour le refroidissement.</p> <p>Sélections possibles :</p> <ul style="list-style-type: none"> • PID (Température) (=par défaut) ("PIDSelfTune.SUT.TuneRuleCool" = 2) ("PIDSelfTune.TIR.TuneRuleCool" = 0) • PID ("PIDSelfTune.SUT.TuneRuleCool" = 0) ("PIDSelfTune.TIR.TuneRuleCool" = 0) • PI ("PIDSelfTune.SUT.TuneRuleCool" = 1) ("PIDSelfTune.TIR.TuneRuleCool" = 4) <p>Toute autre combinaison affiche "Personnalisé" mais "Personnalisé" n'est pas réglé par défaut.</p> <p>"PID (Température)" est nouveau pour PID_Temp, avec une méthode de pré-optimisation (SUT) spécifique pour les processus de température.</p> <p>Disponible seulement si vous cochez/sélectionnez les éléments suivants : "Activer sortie (refroidissement)" dans la vue "Paramètres de base" ("Config.ActivateCooling" = TRUE), et "Commutation des paramètres PID" dans la vue "Paramètres de sortie" (Config.AdvancedCooling</p>

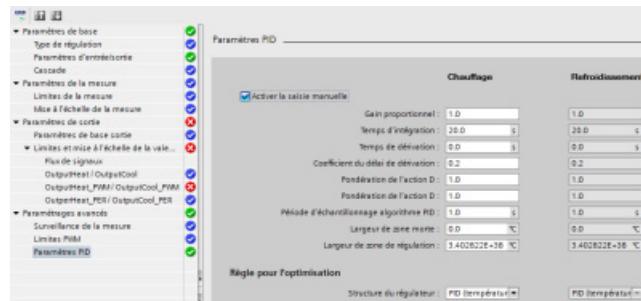
Paramètre	Paramètre TO-DB	Type de données	Plage de valeurs	Description

- 1 Ce champ affiche "s" (secondes) comme unité de temps.
- 2 Disponible seulement si vous cochez "Activer l'entrée manuelle" dans les paramètres PID ("Retain.CtrlParams.SetByUser" = TRUE).
- 3 L'unité de mesure est affichée à la fin du champ comme sélectionné dans la vue "Paramètres de base".
- 4 Disponible seulement si vous cochez/sélectionnez les éléments suivants : "Activer l'entrée manuelle" dans les paramètres PID ("Retain.CtrlParams.SetByUser" = TRUE), "Activer sortie (refroidissement) dans la vue "Paramètres de base" ("Config.ActivateCooling" = TRUE), et "Commutation des paramètres PID" dans la vue "Paramètres de sortie" (Config.AdvancedCooling = TRUE).

Régulation de la valeur de démarrage PID

Vous pouvez éditer les valeurs réelles des paramètres de configuration PID afin que le comportement du régulateur PID puisse être optimisé en mode en ligne.

Ouvrez les "Objets technologiques" pour votre régulateur PID et son objet "Configuration". Pour accéder à la commande de valeur de début, cliquez sur l'"icône lunettes" dans le coin supérieur gauche de la boîte de dialogue :



Vous pouvez maintenant modifier la valeur des paramètres de configuration de votre régulateur PID tel qu'indiqué dans la figure ci-dessous.

Vous pouvez comparer la valeur réelle à la valeur de démarrage (hors ligne) du projet et à la valeur de démarrage (en ligne) API de chaque paramètre. Ceci est nécessaire pour comparer les différences en ligne/hors ligne du bloc de données Objet technologique (TO-DB) et pour être informé des valeurs qui seront utilisées comme valeurs actuelles à la prochaine commutation Arrêt-Démarrage de l'API. De plus, une icône de comparaison donne une indication visuelle pour vous aider à identifier facilement les différences en ligne/hors ligne :



La figure ci-dessus montre l'écran de paramètre PID avec des icônes de comparaison montrant quelles valeurs sont différentes entre les projets en ligne et hors ligne. Une icône verte indique que les valeurs sont les mêmes ; une icône bleue/orange indique que les valeurs sont différentes.

En outre, cliquez sur le bouton paramètre avec la flèche vers le bas pour ouvrir une petite fenêtre qui montre la valeur de démarrage (hors ligne) du projet et sur la valeur de démarrage (en ligne) API de chaque paramètre :



Serveur Web pour une connectivité Internet facile

Le serveur Web offre un accès aux pages Web de données concernant votre CPU et aux données de process contenues dans la CPU. Avec ces pages Web, vous accédez à la CPU (ou au CP compatible avec le Web) avec le navigateur Web de votre PC ou de votre périphérique portable. Les pages Web standard permettent aux utilisateurs autorisés de réaliser ces fonctions et bien d'autres encore :

- modifier l'état de fonctionnement (ARRÊT et MARCHE) de la CPU.
- surveiller et modifier les variables API, les variables de blocs de données et les valeurs d'E/S
- visualiser et télécharger les journaux de données
- visualiser la mémoire tampon de diagnostic de la CPU.
- mettre à jour le firmware de la CPU.

Le serveur Web vous permet également de créer des pages Web personnalisées qui peuvent accéder aux données de la CPU. Vous pouvez développer ces pages avec le logiciel auteur HTML de votre choix. Vous insérez des commandes "AWP" (programmation Web automatique) dans votre code HTML pour accéder aux données dans la CPU.

Vous définissez des utilisateurs et des niveaux de droits pour le serveur Web dans la configuration de l'appareil pour la CPU dans STEP 7.

Exigences concernant le navigateur Web

Le serveur Web accepte les navigateurs Web pour PC suivants :

- Internet Explorer 8.0
- Internet Explorer 9.0
- Mozilla Firefox 17.0.1
- Google Chrome 23.0
- Apple Safari 5.1.7 (Windows)
- Apple Safari 6.0.2 (Mac)

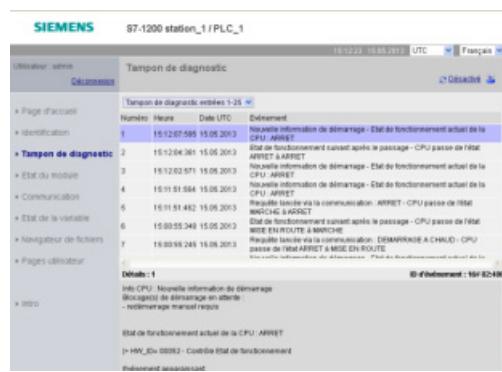
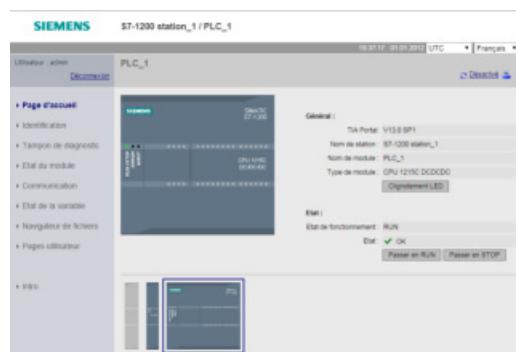
Le serveur Web accepte les navigateurs Web de périphériques mobiles suivants :

- Internet Explorer 6.0 et la version antérieure, pour les pupitres IHM
- Mobile Safari 7534.48.3 (iOS 5.0.1)
- Navigateur Mobile Android 2.3.4
- Mobile Google Chrome 23.0

Les restrictions liées au navigateur, en raison des interférences possibles avec l'affichage de pages Web standard ou personnalisées, sont présentées dans les rubriques sur les contraintes (Page 267).

9.1 Pages Web standard faciles à utiliser

L'utilisation des pages Web standard est simple ! Vous devez seulement activer le Serveur Web lorsque vous configurez la CPU et configurer les utilisateurs du Serveur Web en leur accordant les droits les permettant d'accomplir leurs tâches.



La page de démarrage affiche une représentation de la CPU à laquelle vous êtes connecté et liste les informations générales concernant cette CPU. Si vous avez des CP compatibles avec le Serveur Web, le page Démarrer les affiche également et vous permet de vous connecter à des pages Web via ces CP.

Si vous disposez des droits exigés, vous pouvez modifier l'état de fonctionnement de la CPU (ARRÊT et MARCHE) ou faire clignoter les DEL.

La page État Variable vous permet de contrôler ou modifier toute donnée des E/S ou de la mémoire dans votre CPU. Vous devez disposer du droit "Lire l'état des variables" afin de surveiller des valeurs, et du droit "écrire l'état des variables" afin de modifier des valeurs. Vous pouvez entrer une adresse directe (comme I0.0), un nom de variable API ou une variable depuis un bloc de programme spécifique. Les valeurs de données sont actualisées automatiquement tant que vous ne désactivez pas l'option d'actualisation automatique.

La page Mémoire tampon de diagnostic affiche la mémoire tampon de diagnostic et est accessible aux utilisateurs disposant de droits pour interroger les diagnostics. Vous pouvez sélectionner la plage d'entrées de diagnostic à afficher.

Les entrées de diagnostic listent les événements qui sont survenus et l'heure et la date de la CPU du moment où l'événement est survenu. Sélectionnez un événement pour afficher les informations détaillées le concernant.

La page Navigateur de fichiers vous permet de visualiser, télécharger ou éditer des fichiers dans la mémoire de chargement de la CPU tels que des journaux de données (Page 127) et des recettes. A moins que la CPU n'ait une protection de niveau 4, tous les utilisateurs peuvent visualiser les fichiers à partir de la page Navigateur de fichier. Les utilisateurs possédant des droits de modification de fichiers peuvent supprimer, éditer ou renommer des fichiers.

Outre l'affichage des informations sur les modules sur votre station, la page Informations sur le module vous autorise à mettre à jour la version de firmware sur votre CPU ou d'autres modules qui prennent en charge la mise à jour du firmware. Les utilisateurs possédant des droits pour interroger des diagnostics peuvent visualiser la page des informations sur le module. Les utilisateurs possédant des droits pour effectuer une mise à jour du firmware peuvent mettre à jour le firmware.

D'autres pages Web standard affichent des informations concernant la CPU (comme le numéro de série, la version et le numéro de référence) et les paramètres de communication (comme les adresses réseau ou les propriétés physiques des interfaces de communication).

 **ATTENTION**

Accès non autorisé à la CPU via le serveur Web

L'accès de personnes non autorisées à la CPU ou l'affectation de valeurs invalides aux variables API peut perturber le fonctionnement du processus, ce qui peut entraîner la mort, des blessures graves et des dommages matériels importants.

Comme la validation du serveur Web permet à des utilisateurs autorisés de changer l'état de fonctionnement, d'écrire sur des données de l'API et d'actualiser le firmware, Siemens vous recommande de respecter les consignes sécurité suivantes :

- N'autorisez l'accès au serveur Web qu'avec le protocole HTTPS.
- Protégez les ID d'accès au serveur Web avec un mot de passe fort. Les mots de passe forts contiennent au moins dix caractères, associent des lettres, chiffres et caractères spéciaux, ne sont pas des mots du dictionnaire et ne sont pas des noms ou identifiants pouvant être déduits de vos données personnelles. Ne divulguez pas le mot de passe et changez-en fréquemment.
- N'étendez pas les privilèges minimaux par défaut de l'utilisateur "Everybody".
- Effectuez des recherches d'erreur et des vérifications de plage sur vos variables dans la logique de programme, car les utilisateurs des pages Web ont la possibilité de donner des valeurs invalides aux variables API.
- Utilisez un réseau privé virtuel (VPN) sécurisé pour vous connecter au serveur Web de l'API S7-1200 à partir d'un endroit externe à votre réseau protégé.

9.2 Contraintes qui peuvent affecter l'utilisation du serveur Web

Les considérations informatiques suivantes peuvent influer sur votre utilisation du serveur Web :

- En règle générale, vous devez utiliser l'adresse IP de la CPU ou l'adresse IP d'un routeur sans fil avec un numéro de port pour accéder aux pages Web standard ou aux pages Web personnalisées. Si votre navigateur Web n'autorise pas la connexion directe à une adresse IP, contactez votre administrateur informatique. Si votre réglementation locale prend en charge le service DNS, vous pouvez vous connecter à l'adresse IP via une entrée DNS désignant cette adresse.
- Les pare-feu, les réglages de proxy et autres restrictions spécifiques du site peuvent également restreindre l'accès à la CPU. Contactez votre administrateur informatique pour résoudre ces problèmes.
- Les pages Web standard utilisent des scripts JavaScript et des cookies. Si les scripts JavaScript ou les cookies sont désactivés par les réglages de votre navigateur Web, activez-les. Si vous ne pouvez pas les activer, cela limitera certaines fonctionnalités. L'utilisation de scripts JavaScript et de cookies dans les pages Web personnalisées est facultative. S'ils sont utilisés, vous devez les activer dans votre navigateur.
- Le serveur Web prend en charge le protocole SSL (Secure Sockets Layer). Vous pouvez accéder aux pages Web standard et aux pages Web personnalisées avec une URL <http://ww.xx.yy.zz> ou <https://ww.xx.yy.zz>, "ww.xx.yy.zz" représentant l'adresse IP de la CPU.
- Siemens fournit un certificat de sécurité pour sécuriser l'accès au serveur Web. Vous pouvez télécharger et importer ce certificat dans les options Internet de votre navigateur Web à partir de la page Web standard d'introduction. Si vous décidez de ne pas importer ce certificat, vous recevrez une invitation à vérifier la sécurité à chaque fois que vous accédez au serveur Web avec <https://>.

Nombre de connexions

Le serveur Web accepte 30 connexions HTTP actives au maximum. Diverses actions consomment ces 30 connexions, selon le navigateur Web que vous utilisez et le nombre d'objets différents par page (fichiers .css, images, fichiers .html supplémentaires). Certaines connexions persistent tant que le serveur Web affiche la page alors que d'autres ne persistent pas après la connexion initiale.

Si, par exemple, vous utilisez Mozilla Firefox 8 qui prend en charge six connexions persistantes au maximum, vous pouvez utiliser cinq instances de navigateur ou d'onglet de navigateur avant que le serveur Web ne se mette à couper des connexions. Si une page n'utilise pas les six connexions, vous pouvez avoir des instances de navigateur ou d'onglet de navigateur supplémentaires.

N'oubliez pas que le nombre de connexions actives peut affecter les performances des pages.

Remarque

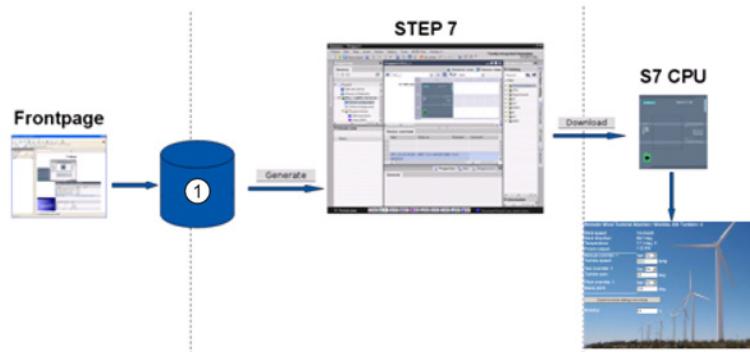
Déconnectez-vous avant de fermer le serveur Web

Si vous êtes connecté au serveur Web, assurez-vous de vous déconnecter avant de fermer le navigateur web. Le serveur Web prend en charge au plus 7 connexions simultanées.

9.3 Pages Web personnalisées faciles à créer

9.3.1 Pages Web "personnalisées" faciles à créer

Le serveur Web du S7-1200 met également à votre disposition la possibilité de créer vos propres pages HTML spécifiques à une application qui incorporent les données depuis l'API. Utilisez l'éditeur HTML de votre choix pour créer ces pages et téléchargez-les ensuite vers la CPU depuis les pages Web standards où elles sont accessibles.



① Fichiers HTML avec commandes AWP intégrées

Ce processus implique plusieurs tâches :

- Créer des pages HTML avec un éditeur HTML
- Inclure des commandes AWP dans les commentaires HTML dans le code HTML : Les commandes AWP forment un ensemble fixe de commandes pour accéder aux informations de la CPU.
- Configurez STEP 7 pour lire et traiter les pages HTML.
- Générez les blocs de programme à partir des pages HTML.
- Programmez STEP 7 pour contrôler l'utilisation des pages HTML.
- Compilez et téléchargez les blocs de programme vers la CPU.
- Accédez aux pages Web personnalisées depuis votre PC ou votre périphérique mobile.

Vous pouvez utiliser le progiciel de votre choix pour créer vos propres pages HTML à utiliser avec le serveur Web. Assurez-vous que le code HTML est conforme aux standards HTML du W3C (World Wide Web Consortium). STEP 7 ne procède à aucune vérification de votre syntaxe HTML.

Vous pouvez utiliser un progiciel qui permet la conception en mode WYSIWYG ou mise en page, mais vous devez pouvoir éditer votre code HTML en HTML pur. La plupart des outils de création Web fournissent ce type d'édition ; sinon, vous pouvez toujours utiliser un éditeur de texte simple pour éditer le code HTML. Insérez la ligne suivante dans votre page HTML pour définir le jeu de caractères de la page à UTF-8 :

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
```

Veillez également à enregistrer le fichier en codage de caractères UTF-8 depuis l'éditeur.

Vous utilisez STEP 7 pour compiler tout le contenu de vos pages HTML en blocs de données STEP 7. Ces blocs de données consistent en un bloc de données de commande qui pilote l'affichage des pages Web et en un ou plusieurs blocs de données de fragment qui contiennent les pages Web compilées. Soyez conscient que les jeux importants de pages HTML, en particulier celles avec de nombreuses images, nécessitent une quantité significative d'espace en mémoire de chargement pour les DB de fragment. Si la mémoire de chargement interne de votre CPU n'est pas suffisante pour vos pages Web personnalisées, utilisez une carte mémoire pour fournir une mémoire de chargement externe.

Pour que votre code HTML utilise des données dans le S7-1200, vous incluez des commandes AWP sous forme de commentaires HTML. Lorsque vous avez terminé, sauvegardez vos pages HTML sur votre PC et notez le chemin du dossier où vous les avez sauvegardées.

Remarque

La taille limite pour les fichiers HTML contenant les commandes AWP est de 64 kilooctets. Vous devez maintenir la taille en dessous de cette limite.

Actualisation de pages Web personnalisées

Les pages Web personnalisées ne sont pas actualisées automatiquement. C'est à vous de choisir si votre programme HTML doit actualiser la page ou non. Le rafraîchissement périodique des pages qui affichent des données de l'API permet d'avoir des données toujours actuelles. Dans les pages HTML servant de formulaires pour la saisie de données, l'actualisation peut interférer avec la saisie des données par l'utilisateur. Si vous voulez que votre page entière soit actualisée automatiquement, vous pouvez ajouter cette ligne dans votre en-tête HTML, "10" étant le nombre de secondes entre les rafraîchissements :

```
<meta http-equiv="Refresh" content="10">
```

Vous pouvez aussi utiliser JavaScript ou d'autres techniques HTML pour gérer l'actualisation de la page ou des données. Consultez la documentation HTML et JavaScript à ce sujet.

9.3.2

Contraintes spécifiques aux pages Web personnalisées

Les contraintes pour les pages Web standard s'appliquent également aux pages Web personnalisées. Il faut en outre tenir compte de quelques points spécifiques pour les pages Web personnalisées.

Espace en mémoire de chargement

Lorsque vous cliquez sur "Générer blocs", vos pages Web personnalisées deviennent des blocs de données qui ont besoin d'espace en mémoire de chargement. Si une carte mémoire est installée, vous disposez de la capacité de votre carte mémoire comme espace de mémoire de chargement externe pour les pages Web personnalisées.

Si vous n'avez pas installé de carte mémoire, ces blocs prennent de l'espace en mémoire de chargement interne qui est limitée selon votre modèle de CPU.

Vous pouvez vérifier la quantité de mémoire de chargement utilisée et disponible à l'aide des outils en ligne et de diagnostic dans STEP 7. Vous pouvez également consulter les propriétés des blocs individuels que STEP 7 génère à partir de vos pages Web personnalisées et voir la mémoire de chargement qu'ils consomment.

Remarque

Si vous devez réduire l'espace nécessaire pour vos pages Web personnalisées, utilisez moins d'images si c'est possible.

Guillemets dans les chaînes de texte

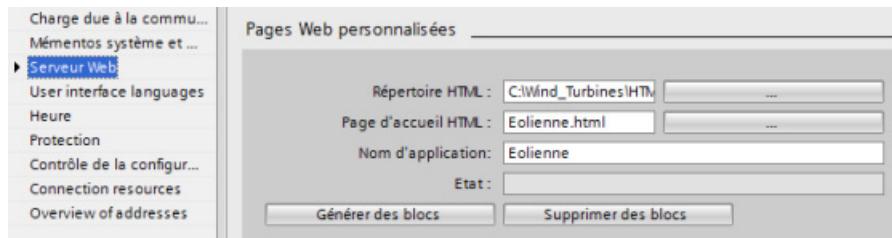
Evitez d'utiliser des chaînes de texte qui contiennent des apostrophes ou des guillemets insérées dans des variables de blocs de données que vous utilisez à n'importe quelle fin dans des pages Web personnalisées. Étant donné que la syntaxe HTML utilise souvent des apostrophes ou des guillemets comme séparateurs, les apostrophes ou les guillemets dans les chaînes de texte peuvent endommager l'affichage des pages Web personnalisées.

Pour les variable de blocs de données de type String que vous utilisez dans les pages Web personnalisées, observez les règles suivantes :

- N'entrez pas d'apostrophes ou de guillemets dans la valeur de la chaîne de variables de blocs de données dans STEP 7.
- Ne laissez pas le programme utilisateur effectuer des affectations de chaînes contenant des apostrophes ou des guillemets dans ces variables de blocs de données.

9.3.3 Configuration d'une page Web personnalisée

Pour configurer les pages Web personnalisées, éditez les propriétés du "serveur Web" de la CPU.



Après avoir activé la fonctionnalité du serveur Web, entrez les informations suivantes :

- Nom et emplacement actuel de la page de démarrage HTML par défaut pour générer les DB pour les pages Web personnalisées.
- Nom de votre application (en option). Le nom de l'application est utilisé pour former des sous-catégories ou groupes de pages Web. Lorsque vous fournissez un nom d'application, le Serveur Web crée une URL pour votre page personnalisée dans le format suivant :

`http[s]://ww.xx.yy.zz/awp/<nom application>/<nom page>.html.`

- Extensions de nom de fichier des fichiers contenant des commandes AWP. Par défaut, STEP 7 analyse les fichiers avec des extensions .htm, .html ou .js. Si vous avez des extensions de fichier supplémentaires, ajoutez-les.
- Numéros d'identification pour le numéro de DB de contrôle et le DB fragmentaire initial

Après avoir configuré le serveur Web, cliquez sur le bouton "Générer blocs" pour générer les DB à partir des pages HTML. Après avoir généré les DB, vos pages Web font partie de votre programme utilisateur. Le bloc de données de contrôle pour le fonctionnement de vos pages Web et les DB "fragmentaires" contiennent l'ensemble des pages HTML.

9.3.4

Utilisation de l'opération WWW

L'opération WWW rend vos pages Web personnalisées accessibles depuis les pages Web standards. Votre programme utilisateur doit exécuter l'instruction WWW une seule fois pour autoriser l'accès aux pages Web définies par l'utilisateur. Il se peut toutefois que vous choisissez de rendre les pages Web définies par l'utilisateur disponibles uniquement dans certaines circonstances. Votre programme utilisateur peut ensuite appeler l'opération WWW selon vos exigences d'application.

Tableau 9- 1 Opération WWW

CONT/LOG	SCL	Description
	<pre>ret_val := #WWW(ctrl_db:=_uint_in_); </pre>	<p>Identifie le DB de commande à utiliser pour les pages Web personnalisées.</p> <p>Le bloc de données de contrôle est le paramètre d'entrée vers l'opération WWW et spécifie le contenu des pages comme représenté dans les blocs de données fragmentaires, ainsi que les informations sur le statut et le contrôle.</p>

Votre programme utilisateur utilise en règle générale le DB de commande directement comme créé par le processus "Générer blocs" sans manipulation supplémentaire. Cependant, le programme utilisateur peut mettre à 1 des commandes globales dans le DB de commande pour désactiver le serveur Web ou pour le réactiver ultérieurement. De plus, pour les pages personnalisées que vous créez comme DB manuels fragmentaires, le programme utilisateur doit contrôler le comportement de ces pages par un tableau de requête dans le DB de commande, comme décrit *Manuel système de l'automate programmable S7-1200*.

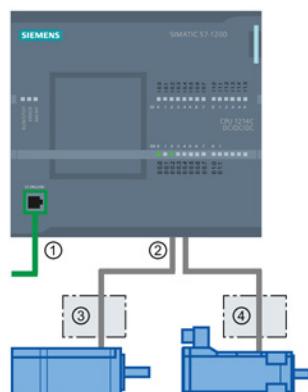
La commande de mouvement est simple

La CPU fournit une fonctionnalité de commande de mouvement (Motion Control) permettant de faire fonctionner des moteurs à pas et des servomoteurs à interface d'impulsion. La fonctionnalité de commande de mouvement se charge de la commande et de la surveillance des entraînements.

- L'objet technologique "Axe" configure les données mécaniques de l'entraînement, l'interface de l'entraînement, les paramètres dynamiques et d'autres propriétés de l'entraînement.
- Vous configurez les sorties d'impulsions et de sens de la CPU pour commander l'entraînement.
- Votre programme utilisateur utilise les instructions de commande de mouvement pour commander l'axe et déclencher des tâches de déplacement.
- Utilisez l'interface PROFINET pour établir la liaison en ligne entre la CPU et la console de programmation. En plus des fonctions en ligne de la CPU, des fonctions supplémentaires de mise en service et de diagnostic sont disponibles pour la commande de mouvement.

Remarque

Les modifications que vous apportez à la configuration de commande de mouvement et que vous chargez à l'état MARCHE ne prennent effet qu'après passage de la CPU de l'état ARRET à l'état MARCHE.



- ① PROFINET
- ② Sorties d'impulsions et de sens
- ③ Partie puissance du moteur pas à pas
- ④ Partie puissance du servomoteur

Les variantes DC/DC/DC de la CPU S7-1200 ont des sorties intégrées pour la commande directe d'entraînements. Les variantes à relais de la CPU ont besoin du Signal Board à sorties DC pour commander des entraînements.

Un Signal Board (SB) augmente les E/S intégrées en ajoutant quelques E/S supplémentaires. Un SB à deux sorties TOR peut être utilisé comme sorties d'impulsions et de sens pour commander un moteur. Un SB à quatre sorties TOR peut être utilisé comme sorties d'impulsions et de sens pour commander deux moteurs. Les sorties relais intégrées ne peuvent pas être utilisées comme sorties d'impulsions pour commander des moteurs. Que vous utilisez les E/S intégrées, les E/S d'un SB ou une combinaison des deux, vous avez au maximum quatre générateurs d'impulsions.

Les entrées et sorties sont affectées par défaut aux quatre générateurs d'impulsions ; toutefois, ils peuvent être affectés à toute sortie TOR sur la CPU ou le SB. Les générateurs d'impulsions sur la CPU ne peuvent pas être affectées à des SM ou des périphéries décentralisées.

Remarque

Les sorties de trains d'impulsions ne peuvent pas être utilisées par d'autres instructions dans le programme utilisateur

Lorsque vous configurez les sorties de la CPU ou du Signal Board en tant que générateurs d'impulsions (pour les instructions PWM ou de commande de mouvement), les adresses des sorties correspondantes ne contrôlent plus les sorties. Si votre programme utilisateur écrit une valeur dans une sortie utilisée comme générateur d'impulsions, la CPU n'écrit pas cette valeur dans la sortie physique.

Tableau 10- 1 Nombre maximum d'entraînements pouvant être commandés

Type de CPU		E/S intégrées ; pas de SB installé		Avec un SB (2 sorties DC)		Avec un SB (4 sorties DC)	
		Avec sens	Sans sens	Avec sens	Sans sens	Avec sens	Sans sens
CPU 1211C	DC/DC/DC	2	4	3	4	4	4
	AC/DC/RLY	0	0	1	2	2	4
	DC/DC/RLY	0	0	1	2	2	4
CPU 1212C	DC/DC/DC	3	4	3	4	4	4
	AC/DC/RLY	0	0	1	2	2	4
	DC/DC/RLY	0	0	1	2	2	4
CPU 1214C	DC/DC/DC	4	4	4	4	4	4
	AC/DC/RLY	0	0	1	2	2	4
	DC/DC/RLY	0	0	1	2	2	4
CPU 1215C	DC/DC/DC	4	4	4	4	4	4
	AC/DC/RLY	0	0	1	2	2	4
	DC/DC/RLY	0	0	1	2	2	4
CPU 1217C	DC/DC/DC	4	4	4	4	4	4

Remarque

Le nombre maximum de générateurs d'impulsions est de quatre.

Que vous utilisez les E/S intégrées, les E/S d'un SB ou une combinaison des deux, vous avez au maximum quatre générateurs d'impulsions.

Tableau 10- 2 Sortie de la CPU : fréquence maximale

CPU	Sortie voie de la CPU	Sortie d'impulsions et de sens	A/B, quadrature, incrément/décrément et impulsion/sens
1211C	Qa.0 à Qa.3	100 kHz	100 kHz
1212C	Qa.0 à Qa.3	100 kHz	100 kHz
	Qa.4, Qa.5	20 kHz	20 kHz
1214C et 1215C	Qa.0 à Qa.3	100 kHz	100 kHz
	Qa.4 à Qb.1	20 kHz	20 kHz
1217C	DQa.0 à DQa.3 (.0+, .0- à .3+, .3-)	1 MHz	1 MHz
	DQa.4 à DQb.1	100 kHz	100 kHz

Tableau 10- 3 Sortie de Signal Board (SB) : fréquence maximale (SB optionnel)

Signal Board (SB)	Sortie voie de SB	Sortie d'impulsions et de sens	A/B, quadrature, incrément/décrément et impulsion/sens
SB 1222, 200 kHz	DQe.0 à DQe.3	200 kHz	200 kHz
SB 1223, 200 kHz	DQe.0, DQe.1	200 kHz	200 kHz
SB 1223	DQe.0, DQe.1	20 kHz	20 kHz

Tableau 10- 4 Fréquences limites des sorties d'impulsions

Sortie d'impulsions	Fréquence
Intégrée	4 PTO : $2 \text{ Hz} \leq f \leq 1 \text{ MHz}$, 4 PTO : $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$, ou la combinaison de ces valeurs pour 4 PTO. ^{1,2}
SB standard	$2 \text{ Hz} \leq f \leq 20 \text{ kHz}$
SB rapides	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$

- 1 Voir le tableau ci-dessous pour quatre combinaisons de fréquence de sorties d'impulsions possibles pour la CPU 1217C.
- 2 Voir le tableau ci-dessous pour quatre combinaisons de fréquence de sorties d'impulsions possibles pour les CPU 1211C, CPU 1212C, CPU 1214C ou CPU 1215C.

Exemple : Configurations de fréquence de sorties d'impulsions pour la CPU 1217C

Remarque

La CPU 1217C peut générer des sorties d'impulsions à des fréquences pouvant atteindre 1 MHz à l'aide des sorties différentielles intégrées.

Les exemples ci-dessous montrent quatre combinaisons de fréquence de sorties d'impulsions possibles :

- Exemple 1 : 4 PTO - 1 MHz, pas de sortie de sens
- Exemple 2 : 1 PTO - 1 MHz, 2 PTO - 100 kHz et 1 PTO - 20 kHz, tous avec sortie de sens
- Exemple 3 : 4 PTO - 200 kHz, pas de sortie de sens
- Exemple 4 : 2 PTO - 100 kHz et 2 PTO - 200 kHz, tous avec sorties de sens

P = Impulsion D = Sens		Sorties intégrées CPU										Sorties SB rapides				Sorties SB standard	
		Sorties 1 MHz (Q)				Sorties 100 kHz (Q)						Sorties 200 kHz (Q)				Sorties 20 kHz (Q)	
		0.0+	0.1+	0.2+	0.3+	0.4	0.5	0.6	0.7	1.0	1.1	4.0	4.1	4.2	4.3	4.0	4.1
		0.0-	0.1-	0.2-	0.3-												
Ex. 1: 4 - 1 MHz (pas de sortie de sens)	PTO1	P															
	PTO2		P														
	PTO3			P													
	PTO4				P												
Ex. 2: 1 PTO - 1 MHz, 2 PTO - 100 et 1 PTO - 20 kHz (tous avec sortie de sens)	PTO1	P	D														
	PTO2					P	D										
	PTO3							P	D								
	PTO4															P	D
Ex. 3: 4 PTO - 200 kHz (pas de sortie de sens)	PTO1											P					
	PTO2											P					
	PTO3											P					
	PTO4											P					
Ex. 4: 2 PTO - 100	PTO1					P	D										
	PTO2							P	D								
	PTO3											P	D				

P = Impulsion D = Sens		Sorties intégrées CPU										Sorties SB rapides			Sorties SB standard		
kHz ; 2 PTO - 200 kHz (tous avec sortie de sens)	PTO4													P	D		

Exemple : Configurations de fréquence de sorties d'impulsions pour les CPU 1211C, CPU 1212C, CPU 1214C et CPU 1215C

Les exemples ci-dessous montrent quatre combinaisons de fréquence de sorties d'impulsions possibles :

- Exemple 1 : 4 PTO - 100 kHz, pas de sortie de sens
- Exemple 2 : 2 PTO - 100 kHz et 2 PTO - 20 kHz, tous avec sorties de sens
- Exemple 3 : 4 PTO - 200 kHz, pas de sortie de sens
- Exemple 4 : 2 PTO - 100 kHz et 2 PTO - 200 kHz, tous avec sorties de sens

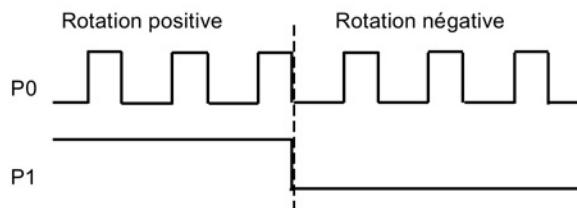
P = Impulsion D = Sens		Sorties intégrées CPU										Sorties SB rapides			Sorties SB lentes		
Ex. 1: 4 PTO - 100 kHz (pas de sortie de sens)		Sorties 100 kHz (Q)				Sorties 20 kHz (Q)						Sorties 200 kHz (Q)				Sorties 20 kHz (Q)	
		0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	1.0	1.1	4.0	4.1	4.2	4.3	4.0	4.1
		CPU 1211C				CPU 1212C						CPU 1214C				CPU 1215C	
		CPU 1211C				CPU 1212C						CPU 1214C				CPU 1215C	
		CPU 1211C				CPU 1212C						CPU 1214C				CPU 1215C	
Ex. 2: 2 PTO - 100	PTO1	P															
	PTO2		P														
	PTO3			P													
Ex. 3: 4 PTO - 200	PTO4				P												
	PTO1	P	D														
	PTO2			P	D												
Ex. 4: 2 PTO - 200	PTO3					P	D										

P = Impulsion D = Sens		Sorties intégrées CPU									Sorties SB rapides			Sorties SB lentes	
kHz ; 2 PTO - 20 kHz (tous avec sor- tie de sens)	PTO4							P	D						
Ex. 3: 4 PTO - 200 kHz (pas de sortie de sens)	PTO1										P				
	PTO2										P				
	PTO3										P				
	PTO4										P				
Ex. 4: 2 PTO - 100 kHz ; 2 PTO - 200 kHz (tous avec sor- tie de sens)	PTO1	P	D												
	PTO2			P	D										
	PTO3									P	D				
	PTO4										P	D			

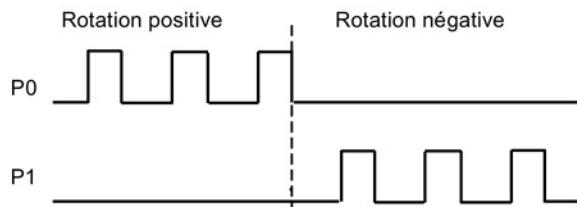
10.1 Phasage

Vous avez quatre options pour l'interface de "phasage" pour l'entraînement par moteur pas-à-pas/servomécanisme. Ces options sont les suivantes :

- PTO (impulsion A et sens B) : Si vous sélectionnez une option de PTO (impulsion A et sens B), alors une sortie (P0) commande la génération d'impulsions et une sortie (P1) commande le sens. P1 est élevé (actif) si la génération d'impulsions est dans le sens positif. P1 est faible (inactif) si la génération d'impulsions est dans le sens négatif :

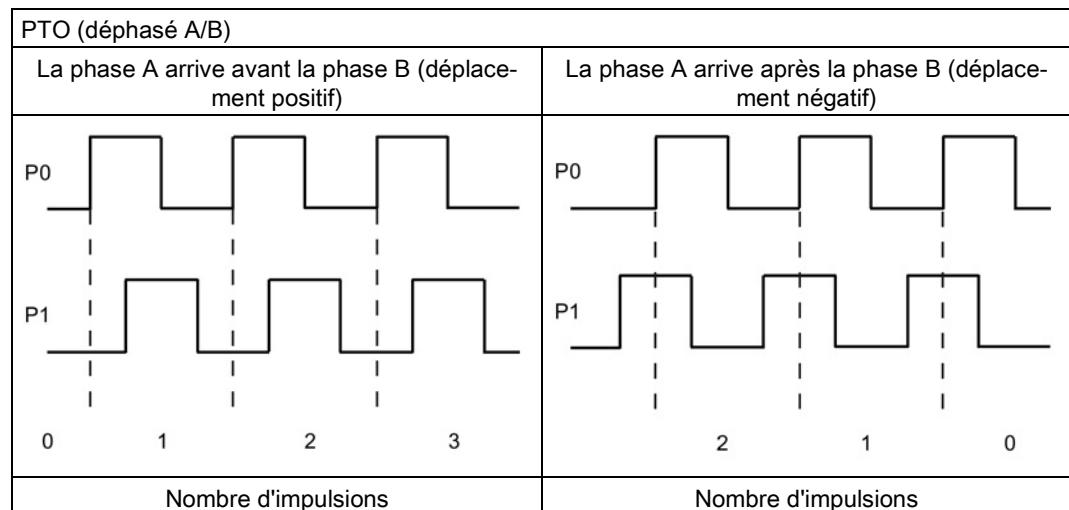


- PTO (comptage ascendant A et comptage descendant B) : Si vous sélectionnez une option PTO (comptage ascendant A et comptage descendant B), alors une sortie (P0) envoie des impulsions pour des sens positifs et une sortie différente (P1) envoie des impulsions pour des sens négatifs :



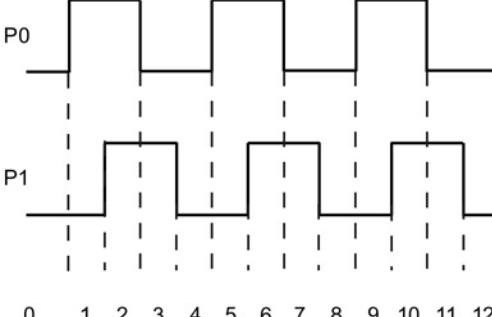
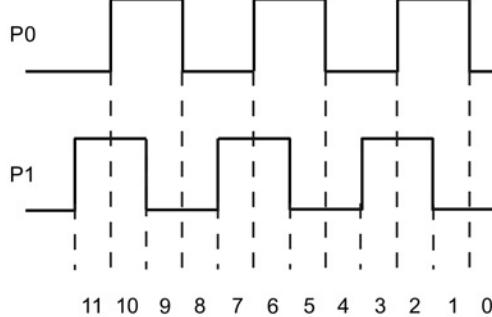
- PTO (déphasé A/B) : Si vous sélectionnez une option PTO (déphasé A/B), alors les deux sorties envoient des impulsions à la vitesse spécifiée, mais avec un déphasage de 90 degrés. C'est une configuration 1X, ce qui signifie qu'une impulsion est la durée entre des commutations positives de P0. Dans ce cas, le sens est déterminé par la sortie qui effectue une commutation élevée la première. P0 arrive avant P1 pour le sens positif. P1 arrive avant P0 pour le sens négatif.

Le nombre d'impulsions générées est fonction du nombre de commutations de 0 à 1 de la phase A. Le rapport de phase détermine le sens de déplacement :



- PTO (déphasé A/B - multiplié par quatre) : Si vous sélectionnez une option PTO (déphasé A/B - multiplié par quatre), alors les deux sorties envoient des impulsions à la vitesse spécifiée, mais avec un déphasage de 90 degrés. La configuration multipliée par quatre est une configuration 4X, ce qui signifie qu'une impulsion est la transition de chaque sortie (positive et négative). Dans ce cas, le sens est déterminé par la sortie qui effectue une commutation élevée la première. P0 arrive avant P1 pour le sens positif. P1 arrive avant P0 pour le sens négatif.

La multiplication par quatre est basée sur des transitions positives et négatives de la phase A et de la phase B. Vous configurez le nombre de transitions. Le rapport de phase (A arrive avant B ou B arrive avant A) détermine le sens de déplacement.

PTO (déphasé A/B - multiplié par quatre)											
La phase A arrive avant la phase B (déplacement positif)						La phase A arrive après la phase B (déplacement négatif)					
											
Nombre d'impulsions						Nombre d'impulsions					

- PTO (impulsion et sens (sens désélectionné)) : Si vous désélectionnez la sortie de sens dans un PTO (impulsion et sens (sens désélectionné)), alors la sortie (P0) commande la génération d'impulsions. La sortie P1 n'est pas utilisée et est disponible pour des utilisations d'autres programmes. Seules des instructions de déplacement positif sont acceptées par la CPU dans ce mode. La commande de déplacement vous contraint à ne pas faire de configurations négatives illégales quand vous sélectionnez ce mode. Vous pouvez enregistrer une sortie si votre application de déplacement est dans un seul sens. La phase unique (une sortie) est représentée dans la figure ci-dessous (en supposant une polarité positive) :



10.2 Configuration d'un générateur d'impulsions

1. Ajoutez un objet technologique :

- Dans l'arborescence du projet, affichez le détail du nœud "Objets technologiques" et sélectionnez "Ajouter nouvel objet".
- Sélectionnez l'icône "Axe" (renommez-la si nécessaire) et cliquez sur "OK" pour ouvrir l'éditeur de configuration pour l'objet axe.
- Affichez les propriétés "Sélection de PTO pour la commande d'axe" sous les "Paramètres de base" et sélectionnez l'impulsion désirée.

Remarque

Si PTO n'a pas été configuré auparavant dans les propriétés CPU, il est configuré pour utiliser l'une des sorties intégrées.

Si vous utilisez une sortie de Signal Board, sélectionnez le bouton "Configuration d'appareil" pour afficher les propriétés CPU. Dans les "Options d'impulsions" sous "Affectation de paramètres", configurez la source de la sortie sur une sortie de Signal Board.

- Configurez les autres paramètres de base et avancés.

2. Programmez votre application : Insérez l'instruction MC_Power dans un bloc de code.

- Pour l'entrée Axis, sélectionnez l'objet technologique axe que vous avez créé et configuré.
- Lorsque l'entrée Enable à la valeur VRAI, les autres instructions de mouvement sont autorisées à fonctionner.
- Lorsque l'entrée Enable a la valeur FAUX, les autres instructions de mouvement sont annulées.

Remarque

N'utilisez qu'une instruction MC_Power par axe.

3. Insérez les autres instructions de mouvement pour produire le mouvement requis.

Remarque

Configuration d'un générateur d'impulsions sur des sorties de Signal Board : Sélectionnez les propriétés "Générateurs d'impulsions (PTO/PWM)" pour une CPU (dans la configuration d'appareil) et activez un générateur d'impulsions. Deux générateurs d'impulsions sont disponibles pour chaque CPU S7-1200 V1.0, V2.0, V2.1 et V2.2. Les CPU S7-1200 V3.0 et V4.0 disposent de quatre générateurs d'impulsions. Sous "Options d'impulsions" dans cette même zone de configuration, sélectionnez "Générateur d'impulsions sous forme de : PTO".

Remarque

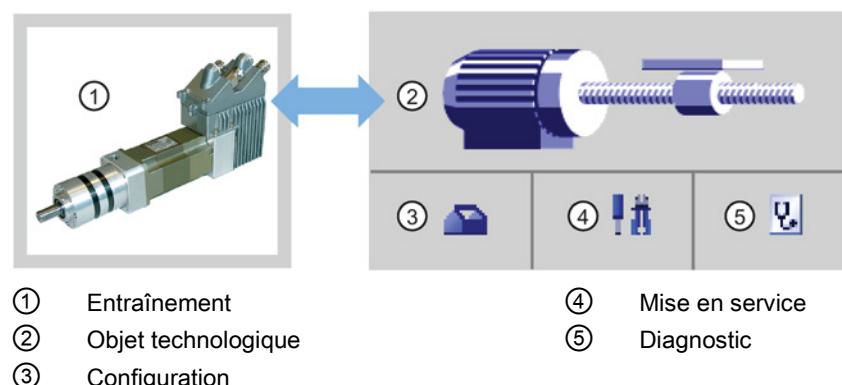
La CPU calcule les tâches de mouvement en "tranches" ou segments de 10 ms. Pendant l'exécution d'une tranche, la tranche suivante est mise en file d'attente pour être exécutée. Si vous interrompez la tâche de mouvement en cours sur un axe (par l'exécution d'une autre tâche de mouvement pour cet axe), l'exécution de cette nouvelle tâche de mouvement pourra ne commencer qu'après 20 ms au maximum (reste de la tranche en cours + tranche en file d'attente).

10.3 Commande de mouvement

10.3.1 Configuration de l'axe

Vous connectez l'axe de la boucle ouverte sur l'API et l'entraînement par le biais d'une PTO (sortie de train d'impulsions).

STEP 7 fournit les outils de configuration, les outils de mise en service et les outils de diagnostic pour l'objet technologique "Axe".

**Remarque**

Pour les versions de firmware CPU V2.2 ou antérieures, PTO exige la fonctionnalité interne d'un compteur rapide (HSC). Cela signifie que le compteur rapide HSC correspondant ne peut pas être utilisé ailleurs.

L'affectation entre PTO et HSC est fixe. Lorsque PTO1 est activé, il est associé à HSC1. Lorsque PTO2 est activé, il est associé à HSC2. Vous ne pouvez pas visualiser la valeur en cours (par exemple, dans ID1000) lorsque des impulsions surviennent.

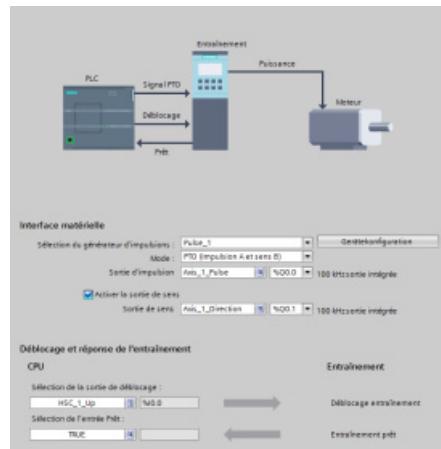
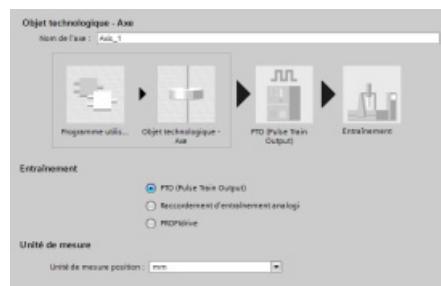
Les CPU S7-1200 V3.0 ou plus ne présentent pas cette restriction ; tous les HSC restent disponibles pour le programme lorsque des sorties d'impulsions sont configurées dans les CPU concernées.

Tableau 10- 5 Outils STEP 7 pour la commande de mouvement

Outil	Description
Configuration	Configure les propriétés suivantes de l'objet technologique "Axe" : <ul style="list-style-type: none">• Sélection de la sortie PTO à utiliser et configuration de l'interface d'entraînement• Propriétés de la mécanique et du rapport de transmission de l'entraînement (ou machine ou système)• Propriétés pour les limites de positionnement, la dynamique et le référencement Sauvegardez la configuration dans le bloc de données de l'objet technologique.
Mise en service	Teste la fonction de votre axe sans avoir à créer un programme utilisateur. Lorsque l'outil est lancé, le panneau de commande s'affiche. Les commandes suivantes sont disponibles sur le panneau de commande : <ul style="list-style-type: none">• Libérer et bloquer l'axe• Déplacer l'axe en mode Manuel à vue• Positionner l'axe en termes absolus et relatifs• Référencer l'axe• Acquitter les erreurs La vitesse et l'accélération/la décélération peuvent être spécifiées pour les commandes de mouvement. Le panneau de commande affiche également l'état en cours de l'axe.
Diagnostic	Surveille l'état en cours et les informations d'erreur concernant l'axe et l'entraînement.

▼ Paramètres de base	<input checked="" type="checkbox"/>
Général	<input checked="" type="checkbox"/>
Entraînement	<input checked="" type="checkbox"/>
▼ Paramètres avancés	<input checked="" type="checkbox"/>
Mécanique	<input checked="" type="checkbox"/>
Limites de position	<input checked="" type="checkbox"/>
▼ Dynamique	<input checked="" type="checkbox"/>
Général	<input checked="" type="checkbox"/>
Arrêt d'urgence	<input checked="" type="checkbox"/>
▼ Référencement	<input checked="" type="checkbox"/>
Activé	<input checked="" type="checkbox"/>
Passif	<input checked="" type="checkbox"/>

Le sélecteur d'arborescence pour l'axe PTO ne comprend pas les menus de configuration Codeur, Modulo, Surveillance de position et Boucle de commande.

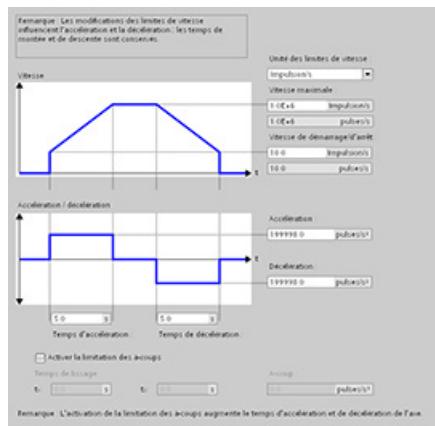


Une fois que vous avez créé l'objet technologique pour l'axe, vous configurez l'axe en définissant les paramètres de base, tels que la sortie PTO et la configuration de l'interface d'entraînement. Vous configurez également d'autres propriétés de l'axe, telles que les limites de positionnement, la dynamique et le référencement.

Remarque

Vous devrez peut-être adapter les valeurs des paramètres d'entrée des instructions de commande de mouvement à la nouvelle unité de dimension dans le programme utilisateur.

10.3 Commande de mouvement

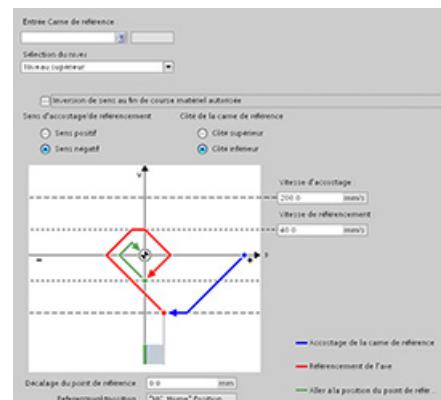
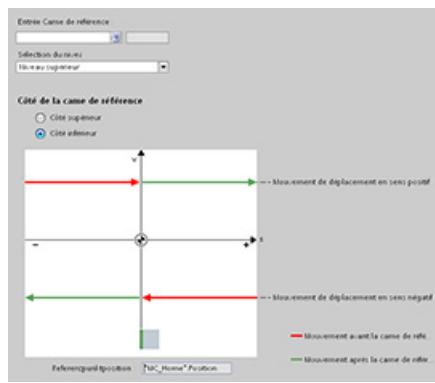


Configurez les propriétés pour les signaux de l'entraînement, la mécanique de l'entraînement et la surveillance de position (fins de course matériels et logiciels).

Vous configurez la dynamique du mouvement et le comportement de la commande d'arrêt d'urgence.



Vous configurez en outre le comportement de référencement (passif et actif).



Utilisez le panneau de commande "Mise en service" pour tester la fonctionnalité indépendamment de votre programme utilisateur.



Cliquez sur l'icône "Démarre" pour mettre l'axe en service.

Le panneau de commande montre l'état en cours de l'axe. Vous pouvez non seulement libérer et bloquer l'axe, mais également tester le positionnement de l'axe (en termes absolus et relatifs) et vous pouvez indiquer la vitesse, l'accélération et la décélération. Vous pouvez également tester les tâches de référencement et le mode Manuel à vue. Le panneau de commande vous permet en outre d'acquitter les erreurs.

10.3.2 Mise en service

Fonction de diagnostic "Bits d'état et d'erreur"

La fonction de diagnostic "Bits d'état et d'erreur" vous permet de surveiller les principaux messages d'état et d'erreur de l'axe. L'affichage de la fonction de diagnostic est disponible en mode en ligne dans les modes de fonctionnement "Commande manuelle" et "Commande automatique" lorsque l'axe est actif.

Tableau 10- 6Etat de l'axe

Etat	Description
Libéré	L'axe est libéré et prêt à être commandé via des tâches de commande de mouvement. (Variable de l'objet technologique : <nom d'axe>.StatusBits.Enable)
Référencé	L'axe est référencé et est en mesure d'exécuter des tâches de positionnement absolu de l'instruction "MC_MoveAbsolute". Il n'est pas nécessaire que l'axe soit référencé pour le positionnement relatif. Situations spéciales : <ul style="list-style-type: none"> Pendant le référencement actif, l'état est FAUX. Si un axe référencé est soumis à un référencement passif, l'état est mis à VRAI pendant le référencement passif. (Variable de l'objet technologique : <nom d'axe>.StatusBits.HomingDone)
Erreur	Une erreur s'est produite dans l'objet technologique Axe. En mode automatique, des informations détaillées sur l'erreur sont fournies par les paramètres ErrorID et ErrorInfo des instructions de commande de mouvement. En mode manuel, le champ "Dernière erreur" du panneau de commande affiche des informations détaillées sur la cause de l'erreur. (Variable de l'objet technologique : <nom d'axe>.StatusBits.Error)
Panneau de commande actif	Le mode "Commande manuelle" a été activé dans le panneau de commande. Le panneau de commande est en charge de la commande de l'objet technologique Axe. L'axe ne peut pas être piloté à partir du programme utilisateur. (Variable de l'objet technologique : <nom d'axe>.StatusBits.ControlPanelActive)

Tableau 10- 7Etat de l'entraînement

Etat	Description
Entraînement prêt	L'entraînement est prêt à fonctionner. (Variable de l'objet technologique : <nom d'axe>.StatusBits.DriveReady)
Erreur	L'entraînement a signalé une erreur après défaillance de son signal Prêt. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.DriveFault)

10.3 Commande de mouvement

Tableau 10- 8 Etat du déplacement de l'axe

Etat	Description
Immobilisation	L'axe est immobilisé. (Variable de l'objet technologique : <nom d'axe>.StatusBits.StandStill)
Accélération	L'axe accélère. (Variable de l'objet technologique : <nom d'axe>.StatusBits.Acceleration)
Vitesse constante	L'axe se déplace à vitesse constante. (Variable de l'objet technologique : <nom d'axe>.StatusBits.ConstantVelocity)
Décélération	L'axe décélère (freine). (Variable de l'objet technologique : <nom d'axe>.StatusBits.Deceleration)

Tableau 10- 9 Etat du mode de déplacement

Etat	Description
Positionnement	L'axe exécute une tâche de positionnement de l'instruction "MC_MoveAbsolute" ou "MC_MoveRelative" ou du panneau de commande. (Variable de l'objet technologique : <nom d'axe>.StatusBits.PositioningCommand)
Prescription de vitesse	L'axe exécute à vitesse prescrite une tâche de l'instruction "MC_MoveVelocity" ou "MC_MoveJog" ou du panneau de commande. (Variable de l'objet technologique : <nom d'axe>.StatusBits.SpeedCommand)
Référencement	L'axe exécute une tâche de référencement de l'instruction "MC_Home" ou du panneau de commande. (Variable de l'objet technologique : <nom d'axe>.StatusBits.Homing)

Tableau 10- 10 Bits d'erreur

Erreur	Description
Fin de course logiciel min. atteint	Le fin de course logiciel inférieur a été atteint. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SwLimitMinReached)
Fin de course logiciel min. dépassé	Le fin de course logiciel inférieur a été dépassé. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SwLimitMinExceeded)
Fin de course logiciel max. atteint	Le fin de course logiciel supérieur a été atteint. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SwLimitMaxReached)
Fin de course logiciel max. dépassé	Le fin de course logiciel supérieur a été dépassé. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SwLimitMaxExceeded)
Fin de course matériel négatif	Le fin de course matériel inférieur a été accosté. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.HwLimitMin)
Fin de course matériel positif	Le fin de course matériel supérieur a été accosté. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.HwLimitMax)
PTO déjà utilisé	Un deuxième axe utilise le même PTO et est activé avec "MC_Power". (Variable de l'objet technologique : <nom d'axe>.ErrorBits.HwUsed)

Erreur	Description
Erreur de configuration	L'objet technologique Axe a été mal configuré ou des données de configuration éditables ont été modifiées de manière incorrecte pendant l'exécution du programme utilisateur. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.ConfigFault)
Erreur générale	Une erreur interne s'est produite. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SystemFault)

Fonction de diagnostic "Etat du déplacement"

La fonction de diagnostic "Etat du déplacement" vous permet de surveiller l'état de déplacement de l'axe. L'affichage de la fonction de diagnostic est disponible en mode en ligne dans les modes de fonctionnement "Commande manuelle" et "Commande automatique" lorsque l'axe est actif.

Tableau 10- 11 Etat du déplacement

Etat	Description
Position cible	Le champ "Position cible" affiche la position cible actuelle d'une tâche de positionnement active de l'instruction "MC_MoveAbsolute" ou "MC_MoveRelative" ou du panneau de commande. La valeur de la position cible n'est valable que pendant l'exécution d'une tâche de positionnement. (Variable de l'objet technologique : <nom d'axe>.MotionStatus.TargetPosition)
Position actuelle	Le champ "Position actuelle" affiche la position actuelle de l'axe. Si l'axe n'est pas référencé, la valeur indique la position par rapport à la position de validation de l'axe. (Variable de l'objet technologique : <nom d'axe>.MotionStatus.Position)
Vitesse actuelle	Le champ "Vitesse actuelle" affiche la vitesse actuelle de l'axe. (Variable de l'objet technologique : <nom d'axe>.MotionStatus.Velocity)

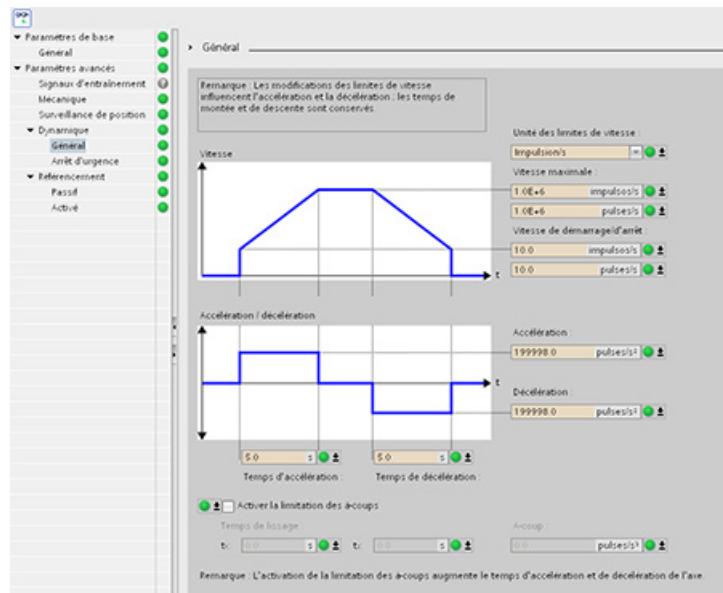
Tableau 10- 12 Limites dynamiques

Limite dynamique	Description
Vitesse	Le champ "Vitesse" affiche la vitesse maximale configurée pour l'axe. (Variable de l'objet technologique : <nom d'axe>.Config.DynamicLimits.MaxVelocity)
Accélération	Le champ "Accélération" affiche l'accélération maximale configurée pour l'axe. (Variable de l'objet technologique : <nom d'axe>.Config.DynamicDefaults.Acceleration)
Décélération	Le champ "Décélération" affiche la décélération maximale configurée pour l'axe. (Variable de l'objet technologique : <nom d'axe>.Config.DynamicDefaults.Deceleration)

Commande de la valeur de début du déplacement

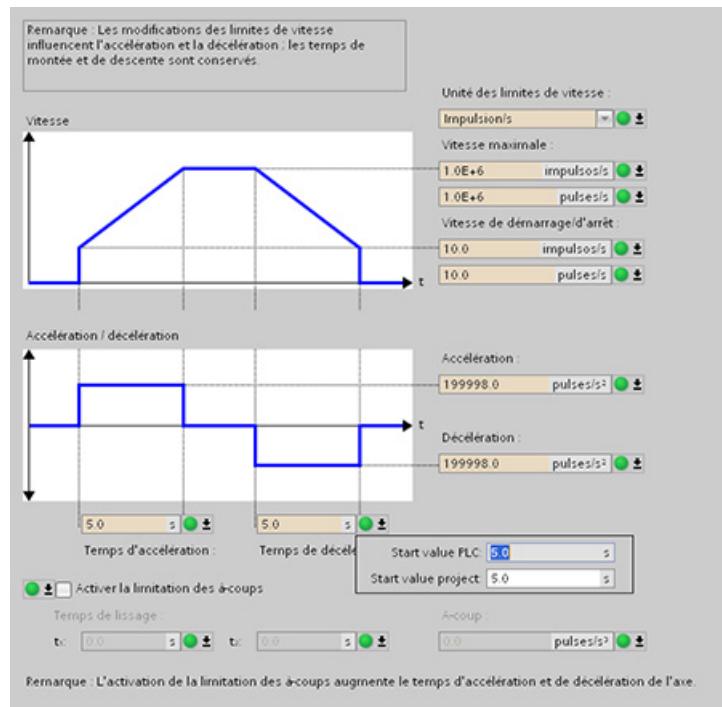
Vous pouvez éditer les valeurs réelles des paramètres de configuration Déplacement afin que le comportement du processus puisse être optimisé en mode en ligne.

Ouvrez les "Objets technologiques" pour votre régulateur de déplacement et son objet "Configuration". Pour accéder à la commande de valeur de début, cliquez sur l'"icône lunettes" dans le coin supérieur gauche de la boîte de dialogue :



Vous pouvez maintenant modifier la valeur de n'importe lequel de vos paramètres de configuration de régulation du déplacement tel qu'indiqué dans la figure ci-dessous.

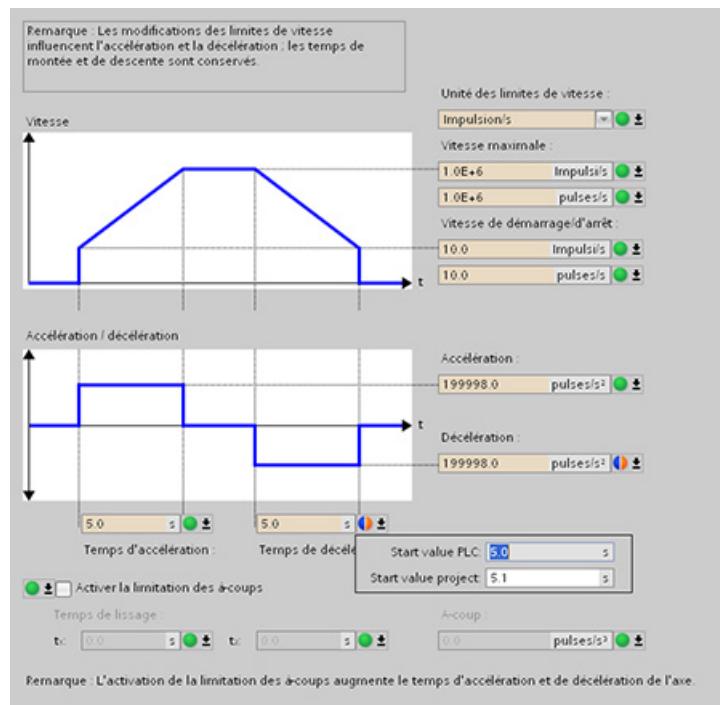
Vous pouvez comparer la valeur réelle à la valeur de démarrage (hors ligne) du projet et à la valeur de démarrage (en ligne) API de chaque paramètre. Ceci est nécessaire pour comparer les différences en ligne/hors ligne du bloc de données Objet technologique (TO-DB) et pour être informé des valeurs qui seront utilisées comme valeurs actuelles à la prochaine commutation Arrêt-Démarrage de l'API. De plus, une icône de comparaison donne une indication visuelle pour vous aider à identifier facilement les différences en ligne/hors ligne.



La figure ci-dessus montre l'écran de paramètre Déplacement avec des icônes de comparaison montrant quelles valeurs sont différentes entre les projets en ligne et hors ligne. Une icône verte indique que les valeurs sont les mêmes ; une icône bleue/verte indique que les valeurs sont différentes.

10.3 Commande de mouvement

En outre, cliquez sur le bouton paramètre avec la flèche vers le bas pour ouvrir une petite fenêtre qui montre la valeur de démarrage (hors ligne) du projet et sur la valeur de démarrage (en ligne) API de chaque paramètre.



10.4 Régulation du mouvement

10.4.1 Configuration de l'axe

Vous connectez l'axe de la boucle fermée sur l'API et l' entraînement par le biais de l' entraînement analogique ou PROFIdrive. L' axe de la boucle fermée requiert également un codeur.

STEP 7 fournit les outils de configuration, les outils de mise en service et les outils de diagnostic pour l' objet technologique "Axe".

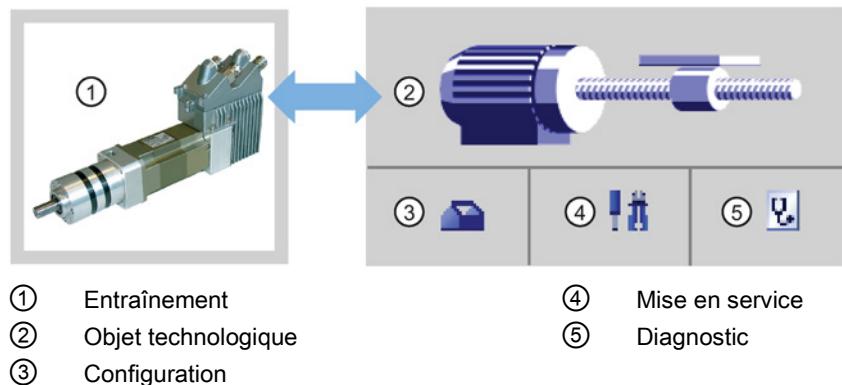


Tableau 10- 13 Outils STEP 7 pour la régulation du mouvement

Outil	Description
Configuration	<p>Configure les propriétés suivantes de l' objet technologique "Axe" :</p> <ul style="list-style-type: none"> • Sélection de la connexion de l' entraînement analogique ou la connexion PROFIdrive à utiliser et configuration de l' interface de l' entraînement et du codeur • Propriétés de la mécanique et du rapport de transmission de l' entraînement et du codeur (ou machine ou système) • Propriétés pour les limites de positionnement, la dynamique et le référencement <p>Sauvegardez la configuration dans le bloc de données de l' objet technologique.</p>
Mise en service	<p>Teste la fonction de votre axe sans avoir à créer un programme utilisateur. Lorsque l' outil est lancé, le panneau de commande s' affiche. Les commandes suivantes sont disponibles sur le panneau de commande :</p> <ul style="list-style-type: none"> • Libérer et bloquer l' axe • Déplacer l' axe en mode Manuel à vue • Positionner l' axe en termes absolus et relatifs • Référencer l' axe • Acquitter les erreurs <p>La vitesse et l' accélération/la décélération peuvent être spécifiées pour les commandes de mouvement. Le panneau de commande affiche également l' état en cours de l' axe.</p>
Diagnostic	Surveille l' état en cours et les informations d' erreur concernant l' axe et l' entraînement.

Remarque

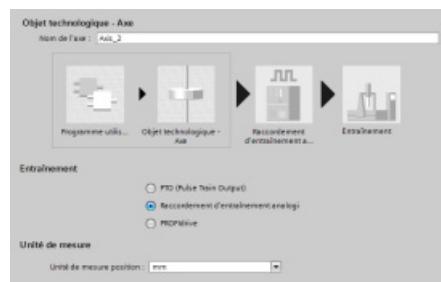
Vous devrez peut-être adapter les valeurs des paramètres d'entrée des instructions de commande de mouvement à la nouvelle unité de dimension dans le programme utilisateur.

Une fois que vous avez créé l'objet technologique pour l'axe, vous configurez l'axe en définissant les paramètres de base, la connexion de l' entraînement analogique ou la connexion PROFIdrive et la configuration de l'interface de l' entraînement et du codeur.

▼ Paramètres de base	✓
Général	✓
Entraînement	✓
Codeur	✓
▼ Paramètres avancés	✓
Mécanique	✓
Modulo	✓
Limites de position	✓
▼ Dynamique	✓
Général	✓
Arrêt d'urgence	✓
▼ Référencement	✓
Activé	✓
Passif	✓
▼ Surveillance de positionnement	✓
Surveillance de positionnement	✓
Écart de traînage	✓
Signal d'immobilisation	✓
Boucle de régulation	✓

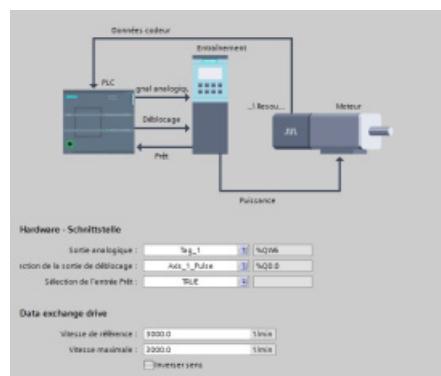
Le sélecteur d'arborescence pour la connexion de l' entraînement analogique ou la connexion PROFIdrive comprend les menus de configuration Codeur, Modulo, Surveillance de position et Boucle de commande.

Configuration de la connexion de l' entraînement analogique



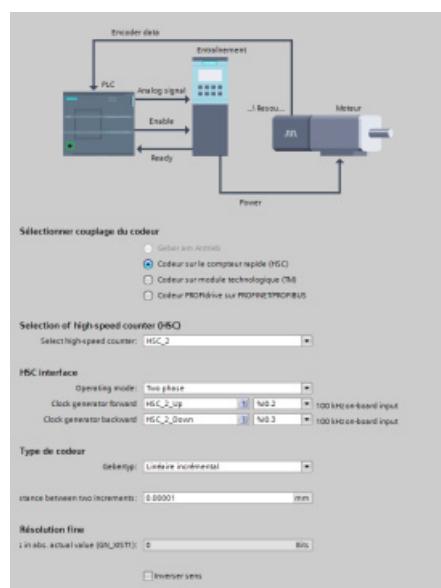
Vous sélectionnez les paramètres suivants dans la boîte de dialogue de configuration générale :

- Bouton radio "Connexion de l' entraînement analogique"
- Unité de mesure



Vous sélectionnez les paramètres suivants dans la boîte de dialogue de configuration de l' entraînement :

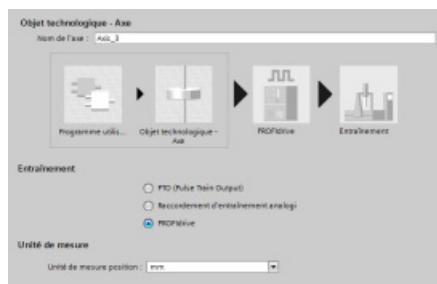
- Sorties matérielles de l' entraînement analogique
- Vitesses d'échange de données de l' entraînement



Vous sélectionnez les paramètres suivants dans la boîte de dialogue de configuration du codeur :

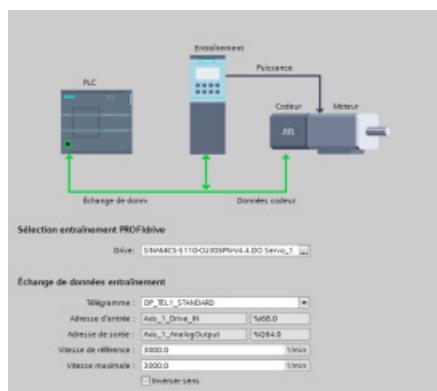
- couplage du codeur de l' entraînement analogique (par exemple un compteur rapide (HSC))
- Interface HSC
- Type de codeur
- Résolution fine

Configuration de PROFIdrive



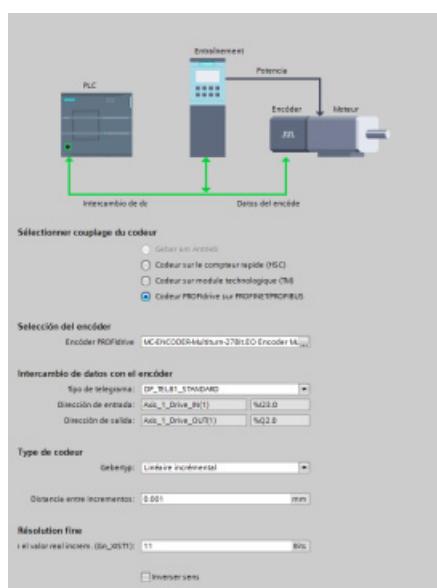
Vous sélectionnez les paramètres suivants dans la boîte de dialogue de configuration générale :

- Bouton radio "PROFIdrive"
- Unité de mesure



Vous sélectionnez les paramètres suivants dans la boîte de dialogue de configuration de l'entraînement :

- Entraînement PROFIdrive
- Echange de données avec l'entraînement



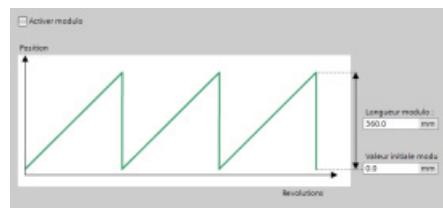
Vous sélectionnez les paramètres suivants dans la boîte de dialogue de configuration du codeur :

- couplage du codeur PROFIdrive (par exemple un codeur PROFIdrive sur PROFINET)
- Codeur PROFIdrive
- Echange de données avec le codeur
- Type de codeur
- Résolution fine

Paramètres avancés

Vous pouvez également configurer les propriétés suivantes de l'axe de la boucle fermée :

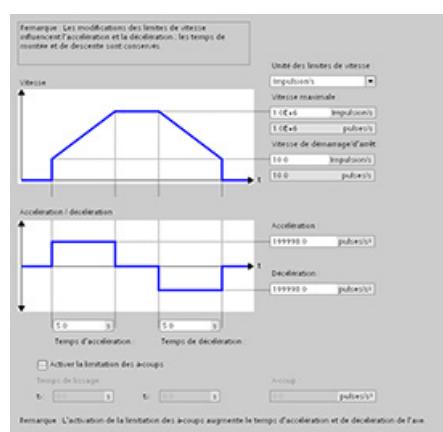
- Modulo
- Limites de position
- Dynamique
- Référencement
- Surveillance de position
- Ecart de traînage
- Signal d'arrêt
- Boucle de régulation



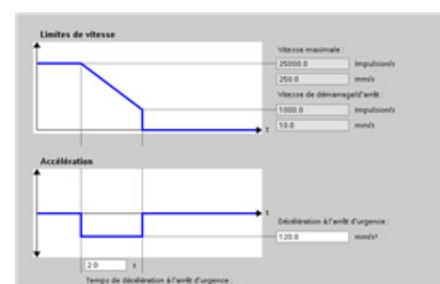
Modulo : Vous pouvez configurer un axe "Modulo" pour déplacer la charge dans une zone cyclique ayant une valeur/position de départ et une longueur définie. Si la position de la charge atteint la fin de cette zone, elle est automatiquement remise à la valeur de départ. Vous activez les champs "Longueur" et "Valeur de départ modulo" lorsque vous cochez la case "Activer modulo".



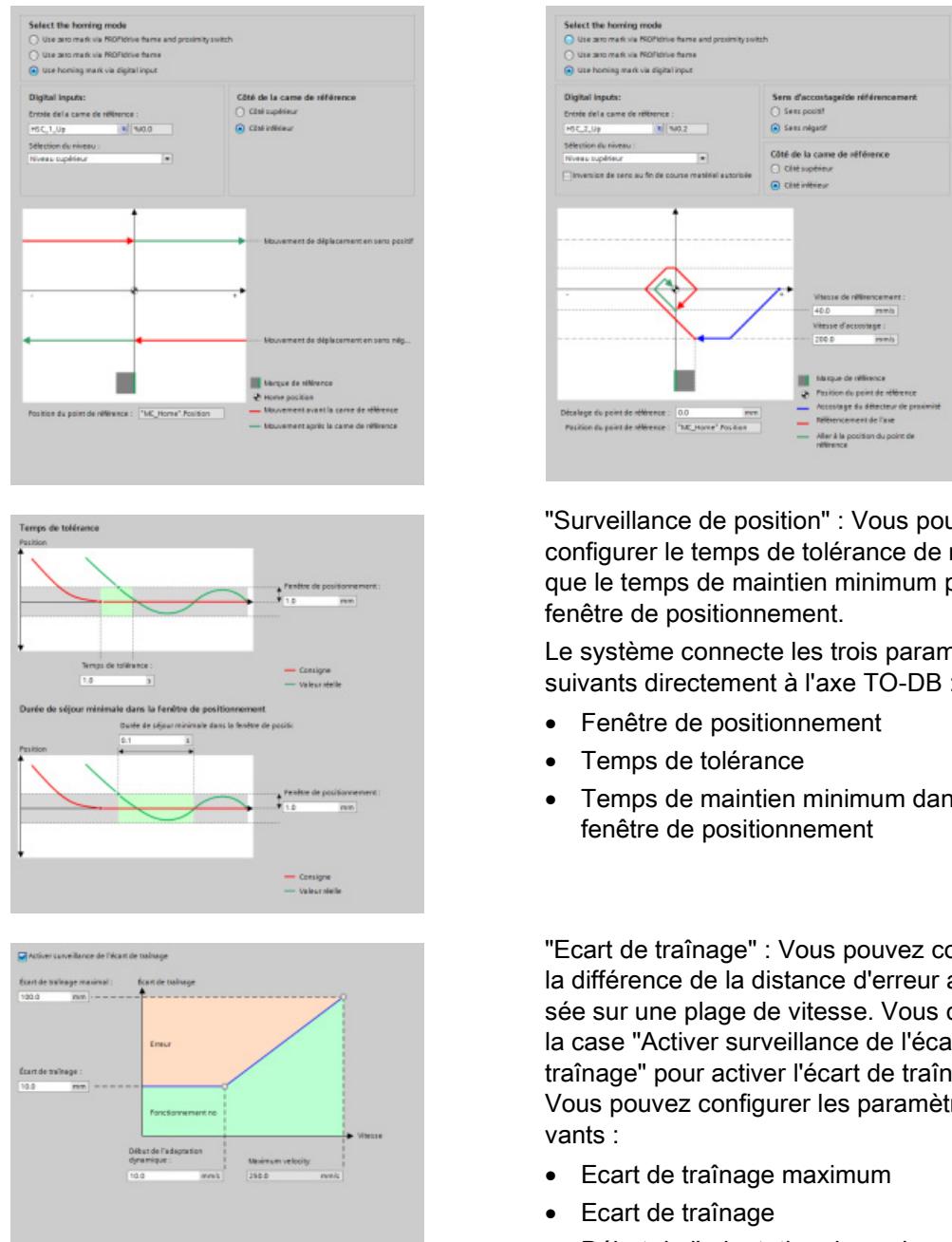
Limites de position : Vous pouvez configurer les propriétés pour les signaux de l'entraînement, la mécanique de l'entraînement et la surveillance de position (fins de course matériels et logiciels).



Dynamique : Vous pouvez configurer la dynamique du mouvement et le comportement de la commande d'arrêt d'urgence.



Référencement : Vous pouvez configurer le comportement de référencement (passif et actif).



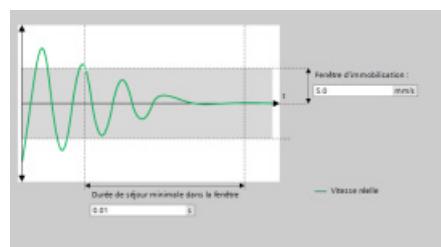
"Surveillance de position" : Vous pouvez configurer le temps de tolérance de même que le temps de maintien minimum pour la fenêtre de positionnement.

Le système connecte les trois paramètres suivants directement à l'axe TO-DB :

- Fenêtre de positionnement
- Temps de tolérance
- Temps de maintien minimum dans la fenêtre de positionnement

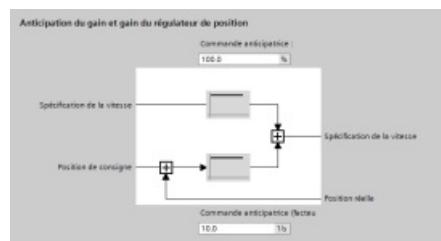
"Ecart de traînage" : Vous pouvez configurer la différence de la distance d'erreur autorisée sur une plage de vitesse. Vous cochez la case "Activer surveillance de l'écart de traînage" pour activer l'écart de traînage. Vous pouvez configurer les paramètres suivants :

- Ecart de traînage maximum
- Ecart de traînage
- Début de l'adaptation dynamique
- Vitesse maximum



"Signal d'arrêt" : Vous pouvez configurer les paramètres suivants :

- Temps de maintien minimum dans la fenêtre d'arrêt
- Fenêtre d'arrêt.



"Boucle de régulation" : Vous pouvez configurer le gain de vitesse dénommé de "Commande anticipatrice (facteur Kv)".

Utilisez le panneau de commande "Mise en service" pour tester la fonctionnalité indépendamment de votre programme utilisateur.



Cliquez sur l'icône "Démarrer" pour mettre l'axe en service.

Le panneau de commande montre l'état en cours de l'axe. Vous pouvez non seulement libérer et bloquer l'axe, mais également tester le positionnement de l'axe (en termes absolus et relatifs) et vous pouvez indiquer la vitesse, l'accélération et la décélération. Vous pouvez également tester les tâches de référencement et le mode Manuel à vue. Le panneau de commande vous permet en outre d'acquitter les erreurs.

10.4.2 Mise en service

Fonction de diagnostic "Bits d'état et d'erreur"

La fonction de diagnostic "Bits d'état et d'erreur" vous permet de surveiller les principaux messages d'état et d'erreur de l'axe. L'affichage de la fonction de diagnostic est disponible en mode en ligne dans les modes de fonctionnement "Commande manuelle" et "Commande automatique" lorsque l'axe est actif.

Tableau 10- 14 Etat de l'axe

Etat	Description
Libéré	L'axe est libéré et prêt à être commandé via des tâches de commande de mouvement. (Variable de l'objet technologique : <nom d'axe>.StatusBits.Enable)
Référencé	L'axe est référencé et est en mesure d'exécuter des tâches de positionnement absolu de l'instruction "MC_MoveAbsolute". Il n'est pas nécessaire que l'axe soit référencé pour le positionnement relatif. Situations spéciales : <ul style="list-style-type: none">• Pendant le référencement actif, l'état est FAUX.• Si un axe référencé est soumis à un référencement passif, l'état est mis à VRAI pendant le référencement passif. (Variable de l'objet technologique : <nom d'axe>.StatusBits.HomingDone)
Erreur	Une erreur s'est produite dans l'objet technologique Axe. En mode automatique, des informations détaillées sur l'erreur sont fournies par les paramètres ErrorID et ErrorInfo des instructions de commande de mouvement. En mode manuel, le champ "Dernière erreur" du panneau de commande affiche des informations détaillées sur la cause de l'erreur. (Variable de l'objet technologique : <nom d'axe>.StatusBits.Error)
Panneau de commande actif	Le mode "Commande manuelle" a été activé dans le panneau de commande. Le panneau de commande est en charge de la commande de l'objet technologique Axe. L'axe ne peut pas être piloté à partir du programme utilisateur. (Variable de l'objet technologique : <nom d'axe>.StatusBits.ControlPanelActive)

Tableau 10- 15 Etat de l'entraînement

Etat	Description
Entraînement prêt	L'entraînement est prêt à fonctionner. (Variable de l'objet technologique : <nom d'axe>.StatusBits.DriveReady)
Erreur	L'entraînement a signalé une erreur après défaillance de son signal Prêt. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.DriveFault)

Tableau 10- 16 Etat du déplacement de l'axe

Etat	Description
Immobilisation	L'axe est immobilisé. (Variable de l'objet technologique : <nom d'axe>.StatusBits.StandStill)
Accélération	L'axe accélère. (Variable de l'objet technologique : <nom d'axe>.StatusBits.Acceleration)
Vitesse constante	L'axe se déplace à vitesse constante. (Variable de l'objet technologique : <nom d'axe>.StatusBits.ConstantVelocity)
Décélération	L'axe décélère (freine). (Variable de l'objet technologique : <nom d'axe>.StatusBits.Deceleration)

Tableau 10- 17 Etat du mode de déplacement

Etat	Description
Positionnement	L'axe exécute une tâche de positionnement de l'instruction "MC_MoveAbsolute" ou "MC_MoveRelative" ou du panneau de commande. (Variable de l'objet technologique : <nom d'axe>.StatusBits.PositioningCommand)
Prescription de vitesse	L'axe exécute à vitesse prescrite une tâche de l'instruction "MC_MoveVelocity" ou "MC_MoveJog" ou du panneau de commande. (Variable de l'objet technologique : <nom d'axe>.StatusBits.SpeedCommand)
Référencement	L'axe exécute une tâche de référencement de l'instruction "MC_Home" ou du panneau de commande. (Variable de l'objet technologique : <nom d'axe>.StatusBits.Homing)

Tableau 10- 18 Bits d'erreur

Erreur	Description
Fin de course logiciel min. atteint	Le fin de course logiciel inférieur a été atteint. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SwLimitMinReached)
Fin de course logiciel min. dépassé	Le fin de course logiciel inférieur a été dépassé. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SwLimitMinExceeded)
Fin de course logiciel max. atteint	Le fin de course logiciel supérieur a été atteint. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SwLimitMaxReached)
Fin de course logiciel max. dépassé	Le fin de course logiciel supérieur a été dépassé. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SwLimitMaxExceeded)
Fin de course matériel négatif	Le fin de course matériel inférieur a été accosté. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.HwLimitMin)
Fin de course matériel positif	Le fin de course matériel supérieur a été accosté. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.HwLimitMax)
PTO déjà utilisé	Un deuxième axe utilise le même PTO et est activé avec "MC_Power". (Variable de l'objet technologique : <nom d'axe>.ErrorBits.HwUsed)

Erreur	Description
Erreur de configuration	L'objet technologique Axe a été mal configuré ou des données de configuration éditables ont été modifiées de manière incorrecte pendant l'exécution du programme utilisateur. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.ConfigFault)
Erreur générale	Une erreur interne s'est produite. (Variable de l'objet technologique : <nom d'axe>.ErrorBits.SystemFault)

Fonction de diagnostic "Etat du déplacement"

La fonction de diagnostic "Etat du déplacement" vous permet de surveiller l'état de déplacement de l'axe. L'affichage de la fonction de diagnostic est disponible en mode en ligne dans les modes de fonctionnement "Commande manuelle" et "Commande automatique" lorsque l'axe est actif.

Tableau 10- 19 Etat du déplacement

Etat	Description
Position cible	Le champ "Position cible" affiche la position cible actuelle d'une tâche de positionnement active de l'instruction "MC_MoveAbsolute" ou "MC_MoveRelative" ou du panneau de commande. La valeur de la position cible n'est valable que pendant l'exécution d'une tâche de positionnement. (Variable de l'objet technologique : <nom d'axe>.MotionStatus.TargetPosition)
Position actuelle	Le champ "Position actuelle" affiche la position actuelle de l'axe. Si l'axe n'est pas référencé, la valeur indique la position par rapport à la position de validation de l'axe. (Variable de l'objet technologique : <nom d'axe>.MotionStatus.Position)
Vitesse actuelle	Le champ "Vitesse actuelle" affiche la vitesse actuelle de l'axe. (Variable de l'objet technologique : <nom d'axe>.MotionStatus.Velocity)

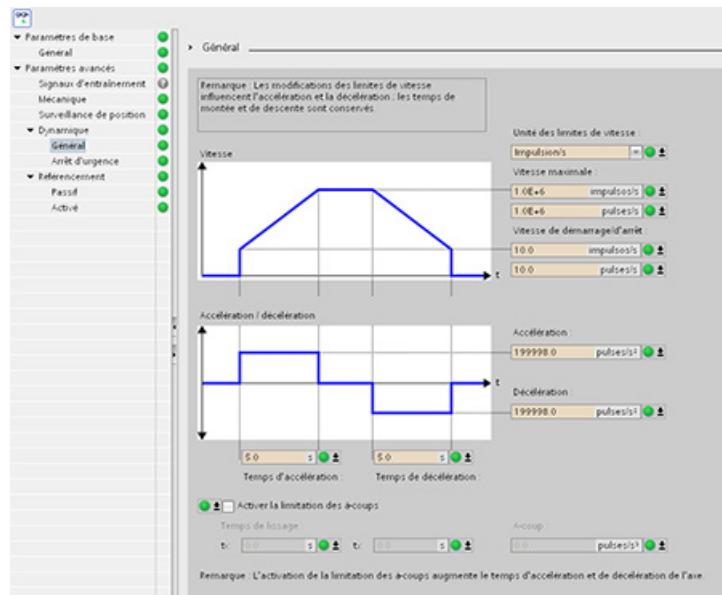
Tableau 10- 20 Limites dynamiques

Limite dynamique	Description
Vitesse	Le champ "Vitesse" affiche la vitesse maximale configurée pour l'axe. (Variable de l'objet technologique : <nom d'axe>.Config.DynamicLimits.MaxVelocity)
Accélération	Le champ "Accélération" affiche l'accélération maximale configurée pour l'axe. (Variable de l'objet technologique : <nom d'axe>.Config.DynamicDefaults.Acceleration)
Décélération	Le champ "Décélération" affiche la décélération maximale configurée pour l'axe. (Variable de l'objet technologique : <nom d'axe>.Config.DynamicDefaults.Deceleration)

Commande de la valeur de début du déplacement

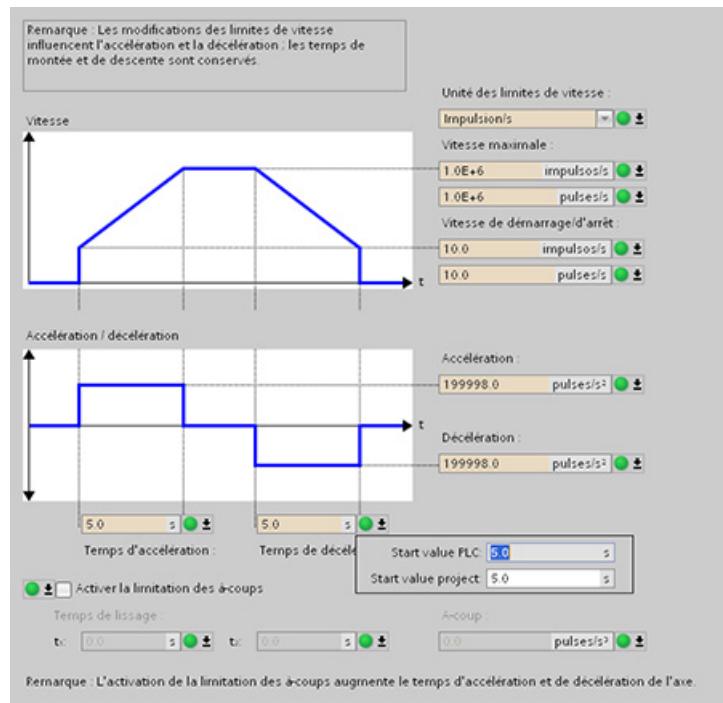
Vous pouvez éditer les valeurs réelles des paramètres de configuration Déplacement afin que le comportement du processus puisse être optimisé en mode en ligne.

Ouvrez les "Objets technologiques" pour votre régulateur de déplacement et son objet "Configuration". Pour accéder à la commande de valeur de début, cliquez sur l'"icône lunettes" dans le coin supérieur gauche de la boîte de dialogue :



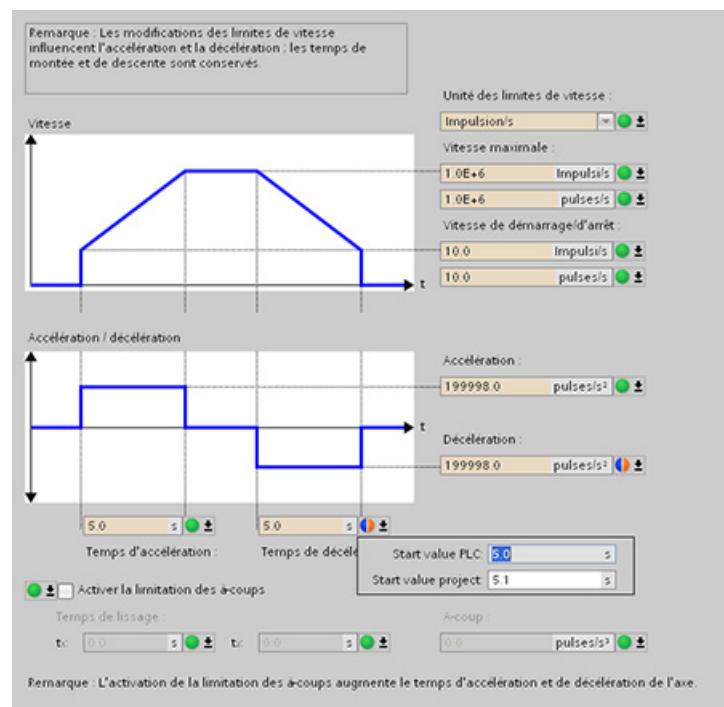
Vous pouvez maintenant modifier la valeur de n'importe lequel de vos paramètres de configuration de régulation du déplacement tel qu'indiqué dans la figure ci-dessous.

Vous pouvez comparer la valeur réelle à la valeur de démarrage (hors ligne) du projet et à la valeur de démarrage (en ligne) API de chaque paramètre. Ceci est nécessaire pour comparer les différences en ligne/hors ligne du bloc de données Objet technologique (TO-DB) et pour être informé des valeurs qui seront utilisées comme valeurs actuelles à la prochaine commutation Arrêt-Démarrage de l'API. De plus, une icône de comparaison donne une indication visuelle pour vous aider à identifier facilement les différences en ligne/hors ligne.



La figure ci-dessus montre l'écran de paramètre Déplacement avec des icônes de comparaison montrant quelles valeurs sont différentes entre les projets en ligne et hors ligne. Une icône verte indique que les valeurs sont les mêmes ; une icône bleue/orange indique que les valeurs sont différentes.

En outre, cliquez sur le bouton paramètre avec la flèche vers le bas pour ouvrir une petite fenêtre qui montre la valeur de démarrage (hors ligne) du projet et sur la valeur de démarrage (en ligne) API de chaque paramètre.

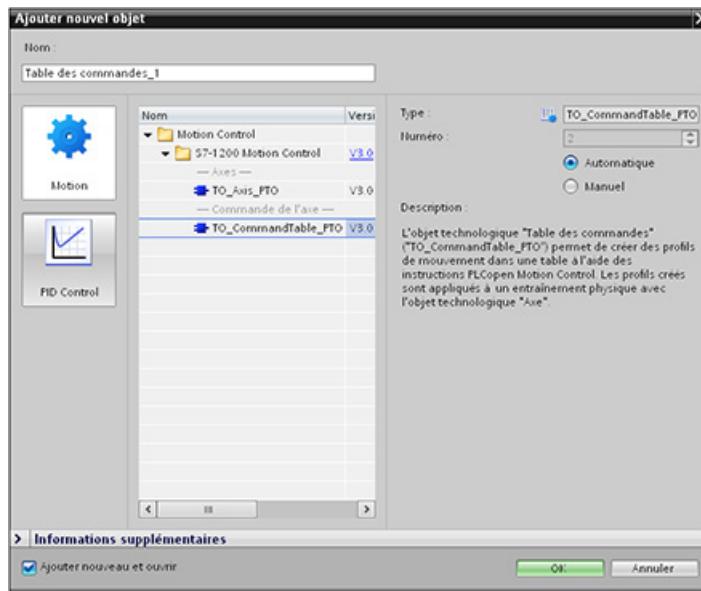


10.5 Configuration de la table de commande *TO_CommandTable_PTO*

Vous pouvez configurer une instruction MC_CommandTable à l'aide des objets technologiques. L'exemple suivant illustre comme cela est possible.

Ajout d'un objet technologique

1. Dans l'arborescence du projet, affichez le détail du nœud "Objets technologiques" et sélectionnez "Ajouter nouvel objet".
2. Sélectionnez l'icône "CommandTable" (renommez-la si nécessaire) et cliquez sur "OK" pour ouvrir l'éditeur de configuration pour l'objet CommandTable.



Planification des étapes pour votre application

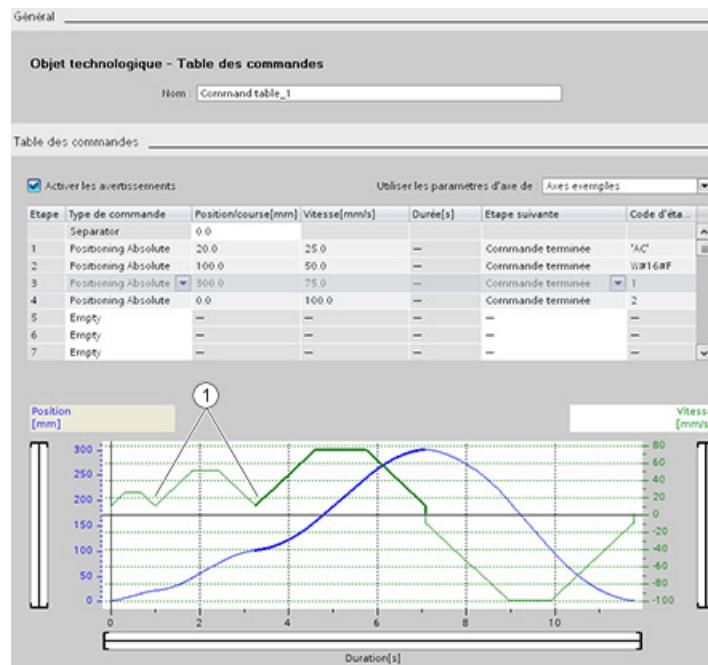
Vous pouvez créer la séquence de mouvement désirée dans la fenêtre de configuration "Table de commande" et vérifier le résultat dans la vue graphique du graphique de tendance.

Vous pouvez sélectionner les types de commandes qui doivent être utilisés pour le traitement de la table de commande. Il est possible d'entrer jusqu'à 32 étapes. Les commandes sont traitées l'une après l'autre, produisant aisément un profil de mouvement complexe.

Tableau 10- 21 Types de commandes MC_CommandTable

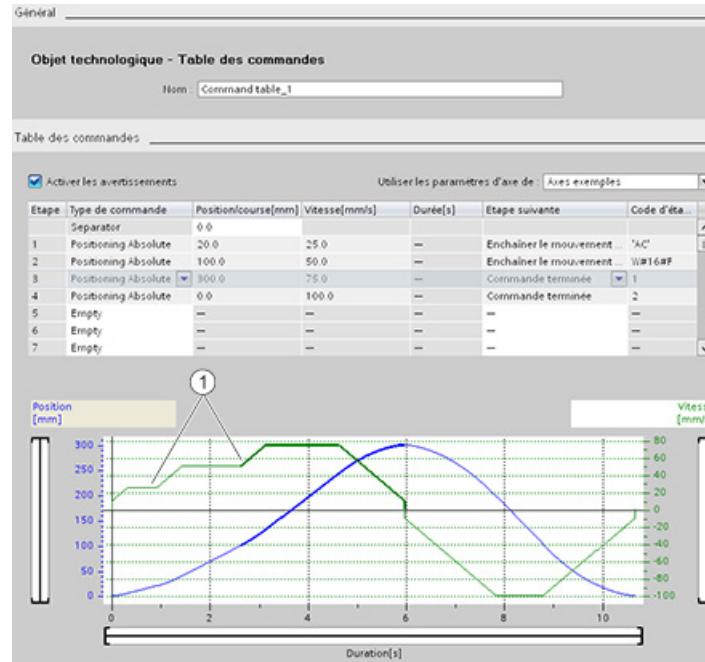
Type de commande	Description
Empty	Marque de réservation pour toute commande à ajouter. L'entrée Empty n'est pas prise en compte lors du traitement de la table de commande.
Halt	Pause de l'axe. Remarque : cette commande n'est exécutée qu'après une commande de consigne de vitesse.
Positioning Relative	Positionnement de l'axe en fonction de la distance. La commande déplace l'axe de la distance indiquée à la vitesse donnée.
Positioning Absolute	Positionnement de l'axe en fonction de l'emplacement. La commande déplace l'axe à l'emplacement indiqué, en utilisant la vitesse donnée.
Velocity setpoint	Déplacement de l'axe à la vitesse donnée.
Wait	Attente de l'expiration de la durée indiquée. "Wait" n'interrompt pas un accostage actif.
Separator	Ajout d'une ligne séparatrice au-dessus de la ligne sélectionnée. La ligne séparatrice permet de définir plus d'un profil dans une seule table de commande.

Dans la figure ci-dessous, "Commande achevée" est utilisé comme transition à l'étape suivante. Ce type de transition permet à votre appareil de décélérer jusqu'à la vitesse de démarrage/d'arrêt, puis de réaccélérer au début de l'étape suivante.



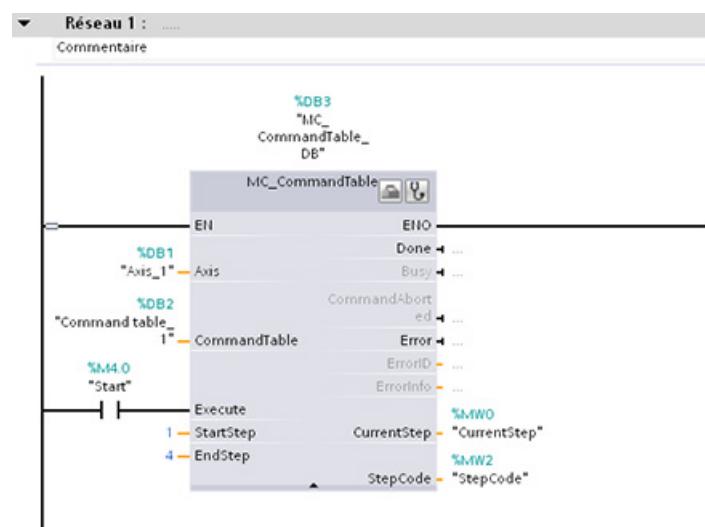
① L'axe décélère jusqu'à la vitesse de démarrage/d'arrêt entre les étapes.

Dans la figure ci-dessous, "Lissage mouvement" est utilisé comme transition à l'étape suivante. Ce type de transition permet à l'appareil de conserver sa vitesse lors de l'entrée dans l'étape suivante, lissant ainsi la transition d'une étape à la suivante pour l'appareil. Le lissage peut réduire le temps total nécessaire à l'exécution d'un profil complet. Sans lissage, l'exécution de cet exemple prend sept secondes. Avec le lissage, le temps d'exécution est réduit d'une seconde pour atteindre un total de six secondes.



- ① L'axe continue à se déplacer et accélère ou décélère selon la vitesse de l'étape suivante, permettant ainsi un gain de temps et une réduction de l'usure mécanique.

Le fonctionnement de votre table CommandTable est géré par une instruction MC_CommandTable, comme illustré ci-dessous :



10.6 Fonctionnement de la commande de mouvement pour le S7-1200

10.6.1 Sorties de la CPU pour la commande de mouvement

La CPU fournit quatre générateurs d'impulsions. Chaque générateur d'impulsions fournit une sortie d'impulsions et une sortie de sens pour la commande d'un entraînement de moteur pas à pas ou d'un entraînement de servomoteur à interface d'impulsion. La sortie d'impulsions fournit à l'entraînement les impulsions nécessaires au mouvement du moteur. La sortie de sens commande le sens de déplacement de l'entraînement.

La sortie PTO génère un signal carré de fréquence variable. La génération d'impulsions est gérée par les informations de configuration et d'exécution fournies via la configuration matérielle et/ou des SFC/SFB.

Selon la sélection de l'utilisateur, lorsque la CPU est à l'état MARCHE, ce sont soit les valeurs stockées dans la mémoire image, soit les sorties de génération d'impulsions qui pilotent les sorties TOR. A l'état ARRET, le générateur PTO ne pilote pas les sorties.

Les sorties CPU intégrées et les sorties d'un Signal Board peuvent être utilisées comme sorties d'impulsions et de sens. Vous choisissez entre sorties CPU intégrées et sorties de Signal Board pendant la configuration d'appareil, sous "Générateurs d'impulsions (PTO/PWM)" dans l'onglet "Propriétés". Seul PTO (Pulse Train Output) s'applique à la commande de mouvement.

Le tableau ci-dessous montre les affectations par défaut des E/S. Toutefois, les quatre générateurs d'impulsions peuvent être configurés pour utiliser n'importe quelles sorties TOR.

Remarque

Les sorties de trains d'impulsions ne peuvent pas être utilisées par d'autres instructions dans le programme utilisateur.

Lorsque vous configurez les sorties de la CPU ou du Signal Board en tant que générateurs d'impulsions (pour les instructions PWM ou de commande de mouvement), les adresses des sorties correspondantes ne contrôlent plus les sorties. Si votre programme utilisateur écrit une valeur dans une sortie utilisée comme générateur d'impulsions, la CPU n'écrit pas cette valeur dans la sortie physique.

Remarque

Les sorties de sens PTO peuvent être libérées afin d'être utilisées à un autre endroit dans le programme.

Chaque PTO requiert l'affectation de deux sorties : l'une comme sortie d'impulsions et l'autre comme sortie de sens. Il se peut que vous utilisez uniquement la sortie d'impulsions et pas la sortie de sens. Dans ce cas, vous pouvez libérer la sortie de sens pour une autre utilisation dans votre programme utilisateur. La sortie ne peut pas être utilisée simultanément comme sortie de sens PTO et dans le programme utilisateur.

Tableau 10- 22 Affectations par défaut des sorties d'impulsions et de sens

Utilisation des sorties pour la commande de mouvement		
	Impulsion	Sens
PTO1		
E/S intégrées	Q0.0	Q0.1
E/S du SB	Q4.0	Q4.1
PTO2		
E/S intégrées	Q0.2	Q0.3
E/S du SB	Q4.2 ¹	Q4.3 ¹
PTO3		
E/S intégrées	Q0.4 ²	Q0.5 ²
E/S du SB	Q4.0	Q4.1
PTO4		
E/S intégrées	Q0.6 ³	Q0.7 ³
E/S du SB	Q4.2	Q4.3

¹ Les sorties Q4.2 et Q4.3 ne sont disponibles que sur le SB1222 DQ4.

² La CPU 1211C ne comporte pas de sorties Q0.4, Q0.5, Q0.6 et Q0.7. Ces sorties ne peuvent donc pas être utilisées dans la CPU 1211C.

³ La CPU 1212C ne comporte pas de sorties Q0.6 et Q0.7. Ces sorties ne peuvent donc pas être utilisées dans la CPU 1212C.

⁴ Ce tableau s'applique aux fonctions PTO des CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C et CPU 1217C.

Interface d'entraînement

Pour la commande de mouvement, vous pouvez configurer facultativement une interface d'entraînement pour "Entraînement libéré" et "Entraînement prêt". Lors de l'utilisation de l'interface d'entraînement, la sortie TOR pour la validation de l'entraînement et l'entrée TOR pour "Entraînement prêt" peuvent être sélectionnées librement.

Remarque

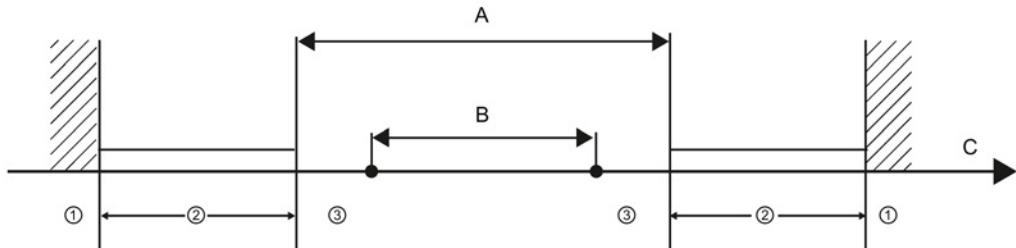
Le firmware prendra le contrôle par le biais des sorties d'impulsions et de sens correspondantes si la sortie PTO (Pulse Train Output) a été sélectionnée et affectée à un axe.

Avec cette mainmise de la fonction de commande, le lien entre la mémoire image et la sortie physique est également rompu. Alors que l'utilisateur peut écrire dans la mémoire image des sorties d'impulsions et de sens via le programme utilisateur ou la table de visualisation, ces valeurs ne sont jamais transférées à la sortie physique. En conséquence, il n'est pas non plus possible de surveiller la sortie physique via le programme utilisateur ou la table de visualisation. Les informations lues ne font que refléter la valeur de la mémoire image et ne correspondent en rien à l'état actuel de la sortie physique.

Pour toutes les autres sorties CPU qui ne sont pas utilisées en permanence par le firmware CPU, l'état de la sortie physique peut être commandé ou surveillé comme d'habitude via la mémoire image.

10.6.2 Fins de course matériels et logiciels pour la commande de mouvement

Utilisez les fins de course matériels et logiciels pour limiter la plage de déplacement autorisée et la plage de travail de votre axe.



- | | | | |
|---|---|---|---|
| ① | Butée mécanique | A | Plage de déplacement autorisée pour l'axe |
| ② | Fins de course matériels inférieur et supérieur | B | Plage de travail de l'axe |
| ③ | Fins de course logiciels inférieur et supérieur | C | Distance |

Il faut activer les fins de course matériels et logiciels avant de les utiliser dans la configuration ou dans le programme utilisateur. Les fins de course logiciels sont actifs uniquement après le référencement de l'axe.

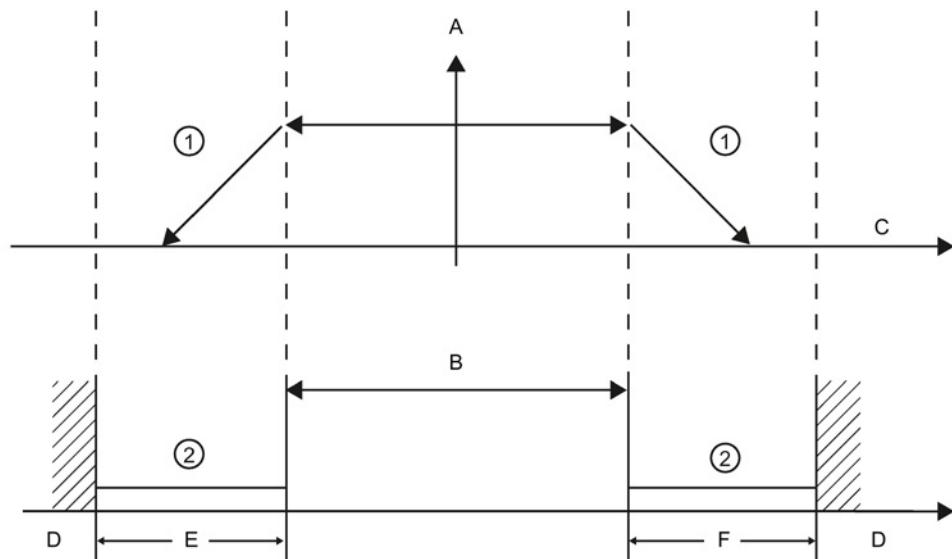
Fins de course matériels

Les fins de course matériels définissent la plage de déplacement maximale de l'axe. Les fins de course matériels sont des commutateurs physiques qui doivent être raccordés à des entrées aptes aux alarmes de la CPU. Utilisez uniquement des fins de course matériels qui restent commutés en permanence après avoir été accostés. Cet état de commutation ne peut être annulé qu'après un retour dans la plage de déplacement autorisée.

Tableau 10- 23 Entrées disponibles pour les limites matérielles

Description	RPS	LIM-	LIM+
E/S intégrées		I0.0 - I1.5	
E/S du SB		I4.0 - I4.3	

Lors de l'accostage des fins de course matériels, l'axe freine jusqu'à s'immobiliser avec la décélération d'urgence configurée. La décélération d'urgence configurée doit être suffisante pour arrêter l'axe de manière fiable avant la butée mécanique. Le schéma suivant montre le comportement de l'axe après qu'il a accosté les fins de course matériels.



- ① L'axe freine jusqu'à s'immobiliser avec la décélération d'urgence configurée.
 - ② Plage dans laquelle les fins de course matériels signalent l'état "accosté".
- A [vitesse]
 B Plage de déplacement autorisée
 C Distance
 D Butée mécanique
 E Fin de course matériel inférieur
 F Fin de course matériel supérieur

ATTENTION

Risques liés à la modification du temps de filtre pour une voie d'entrée TOR

Si le temps de filtre pour une voie d'entrée TOR est modifié par rapport à un paramétrage précédent, une nouvelle valeur d'entrée de niveau "0" peut devoir être présente pendant une durée cumulée allant jusqu'à 20,0 ms pour que le filtre réagisse pleinement aux nouvelles entrées. Pendant ce temps, les événements d'impulsion "0" courts de moins de 20,0 ms peuvent ne pas être détectés ni comptés.

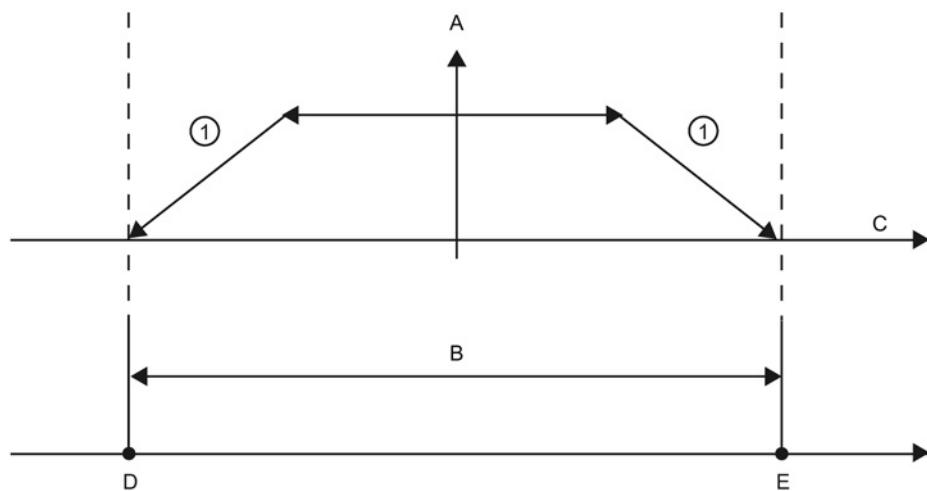
Cette modification des temps de filtre peut provoquer un fonctionnement inattendu des machines ou du processus, pouvant entraîner la mort, des blessures graves et des dommages matériels importants.

Pour garantir la prise en compte immédiate d'un nouveau temps de filtre, mettez la CPU hors tension puis sous tension.

Fins de course logiciels

Les fins de course logiciels limitent la plage de travail de l'axe. Ils doivent se trouver en deçà des fins de course matériels par rapport à la plage de déplacement. Comme les positions des fins de course logiciels sont flexibles, la plage de travail de l'axe peut être restreinte sur une base individuelle en fonction du profil de déplacement en cours. Contrairement aux fins de course matériels, les fins de course logiciels sont réalisés exclusivement au moyen du logiciel et ne nécessitent pas leurs propres commutateurs.

Si des fins de course logiciels sont activés, un déplacement actif est arrêté à la position du fin de course logiciel. L'axe freine avec la décélération configurée. Le schéma suivant montre le comportement de l'axe lorsqu'il atteint les fins de course logiciels.



① L'axe freine jusqu'à s'immobiliser avec la décélération configurée.

A [vitesse]

B Plage de travail

C Distance

D Fin de course logiciel inférieur

E Fin de course logiciel supérieur

Utilisez des fins de course matériels supplémentaires si une butée mécanique est placée après les fins de course logiciels et qu'il y a risque de dommages mécaniques.

Informations supplémentaires

Votre programme utilisateur peut annuler les limites de positionnement matérielles ou logicielles en activant ou en désactivant à la fois la fonctionnalité de limite matérielle et logicielle. La sélection se fait dans le DB de l'axe.

- Pour activer ou désactiver la fonctionnalité de limite matérielle, accédez à la variable "Active" (Bool) dans le chemin de DB "<nom d'axe>/Config/**PositionLimits_HW**". L'état de la variable "Active" active ou désactive l'utilisation de limites de positionnement matérielles.
- Pour activer ou désactiver la fonctionnalité de limite de positionnement logicielle, accédez à la variable "Active" (Bool) dans le chemin de DB "<nom d'axe>/Config/**Position Limits_SW**". L'état de la variable "Active" active ou désactive les limites de positionnement logicielles.

Vous pouvez également modifier les limites de positionnement logicielles avec votre programme utilisateur (par exemple, pour offrir une plus grande souplesse pour la configuration d'une machine ou réduire la durée de remplacement d'une machine). Votre programme utilisateur peut écrire de nouvelles valeurs dans les variables "MinPosition" et "MaxPosition" (unités physiques en format Real) dans le DB <nom d'axe>/Config/**PositionLimits_SW**".

10.6.3 Référencement

10.6.3.1 Référencement de l'axe

Le référencement consiste à établir une correspondance entre les coordonnées de l'axe et la position physique réelle de l'entraînement (si l'entraînement est actuellement en position x, l'axe sera ajusté pour être en position x). Pour les axes commandés en position, les entrées et les affichages concernant la position se réfèrent exactement à ces coordonnées de l'axe.

Remarque

La correspondance entre les coordonnées de l'axe et la situation réelle est extrêmement importante. Cette étape est nécessaire pour garantir que la position cible absolue de l'axe est également atteinte exactement avec l'entraînement.

L'instruction MC_Home déclenche le référencement de l'axe.

Il existe quatre fonctions de référencement différentes. Les deux premières fonctions permettent à l'utilisateur de définir la position en cours de l'axe et les deux autres à positionner l'axe par rapport à un capteur de référencement.

- Mode 0 - Référencement direct absolu : Lorsqu'il est activé, ce mode indique à l'axe exactement où il est. Il donne à la variable de position interne la valeur de l'entrée Position de l'instruction de référencement. Ce mode est utilisé pour le calibrage et la configuration des machines.

La position de l'axe est définie sans tenir compte de la came de référence. Les déplacement en cours ne sont pas annulés. La valeur du paramètre d'entrée Position de l'instruction MC_Home est immédiatement prise comme point de référence de l'axe. Pour affecter le point de référence à une position mécanique précise, l'axe doit être immobile à cette position pendant l'opération de référencement.

- Mode 1 - Référencement direct relatif : Lorsqu'il est activé, ce mode utilise la variable de position interne à laquelle il ajoute la valeur de l'entrée Position de l'instruction de référencement. Ce mode sert typiquement à prendre en compte le décalage d'une machine.

La position de l'axe est définie sans tenir compte de la came de référence. Les déplacement en cours ne sont pas annulés. La position de l'axe après le référencement est définie comme suit : nouvelle position de l'axe = position en cours de l'axe + valeur du paramètre Position de l'instruction MC_Home

- Mode 2 - Référencement passif : Lorsque l'axe se déplace et franchit la came de référence, la position en cours est définie comme position de référence. Cette fonction permet de prendre en compte l'usure normale des machines et le jeu entre dents et d'éviter ainsi la compensation manuelle de l'usure. L'entrée Position de l'instruction de référencement s'ajoute, comme précédemment, à la position indiquée par la came de référence, ce qui permet un décalage aisée de la position de référence.

Lors du référencement passif, l'instruction MC_Home n'exécute aucune prise de référence. L'accostage de la came de référence doit être réalisé par l'utilisateur au moyen d'instructions de commande de mouvement. L'axe est référencé conformément à la configuration lorsque la came de référence est détectée. Les déplacements en cours ne sont pas annulés au démarrage du référencement passif.

- Mode 3 - Référencement actif : Ce mode constitue la méthode de référencement de l'axe la plus précise. Le sens et la vitesse de déplacement initiaux sont configurés dans les paramètres avancés de référencement de la configuration d'objet technologique. Cela dépend de la configuration de la machine. Il est également possible de déterminer si le front montant ou le front descendant du signal de came de référence constitue la position de référence. Pratiquement tous les capteurs ont une plage active et si la position Régime permanent activé était utilisée comme signal de référence, il y aurait la possibilité d'une erreur dans la position de référence puisque la plage active du signal activé couvrirait une plage de distance. L'utilisation du front montant ou du front descendant de ce signal permet d'obtenir des résultats beaucoup plus précis pour la position de référence. Comme avec tous les autres modes, la valeur de l'entrée Position de l'instruction de référencement est ajoutée à la position référencée matérielle.

En mode de référencement actif, l'instruction MC_Home exécute la prise de référence requise. L'axe est référencé conformément à la configuration lorsque la came de référence est détectée. Les déplacement en cours sont annulés.

Les modes 0 et 1 ne nécessitent aucun déplacement de l'axe. Ils servent typiquement lors de la configuration et du calibrage. Les modes 2 et 3 nécessitent un déplacement de l'axe et franchissent un capteur qui est configuré en tant que came de référence dans l'objet technologique Axe. Le point de référence peut être placé dans la zone de travail de l'axe ou hors de la zone de travail normale mais à l'intérieur de la plage de déplacement.

10.6.3.2 Configuration des paramètres de référencement

Configurez les paramètres pour le référencement actif et passif dans la fenêtre de configuration "Référencement". La méthode de référencement est définie à l'aide du paramètre d'entrée "Mode" de l'instruction de commande de mouvement. Ici, Mode = 2 signifie référencement passif et Mode = 3 signifie référencement actif.

Remarque

Prenez l'une des mesures suivantes pour garantir que la machine n'atteint pas une butée mécanique en cas d'inversion de sens :

- Maintenez une vitesse d'accostage faible.
- Augmentez l'accélération/décélération configurée.
- Augmentez la distance entre le fin de course matériel et la butée mécanique.

Tableau 10- 24 Paramètres de configuration pour le référencement de l'axe

Paramètre	Description
Entrée de la came de référence (référencement actif et passif)	<p>Sélectionnez, dans la liste déroulante, l'entrée TOR pour la came de référence. L'entrée doit pouvoir émettre des alarmes. Les entrées intégrées de la CPU et les entrées d'un Signal Board inséré peuvent être sélectionnées comme entrées pour la came de référence.</p> <p>Le temps de filtre par défaut pour les entrées TOR est de 6,4 ms. Lorsqu'on utilise les entrées TOR comme came de référence, il peut se produire des décélérations indésirables et donc des imprécisions. Selon la vitesse d'approche et l'étendue de la came de référence, le point de référence peut ne pas être détecté. Il est possible de régler le temps de filtre sous "Filtre d'entrée" dans la configuration d'appareil des entrées TOR.</p> <p>Le temps de filtre indiqué doit être inférieur à la durée du signal d'entrée au niveau de la came de référence.</p>
Inversion automatique du sens lorsque les fins de course matériels sont atteints (référencement actif uniquement)	<p>Cochez cette case si vous souhaitez utiliser les fins de course matériels comme came d'inversion pour la prise de référence. Les fins de course matériels doivent être configurés et activés pour l'inversion de sens.</p> <p>Si le fin de course matériel est atteint pendant que le référencement est actif, l'axe freine avec la décélération configurée (et non avec la décélération d'urgence) et effectue une inversion de sens. La came de référence est ensuite recherchée en sens inverse.</p> <p>Si l'inversion de sens n'est pas active et que l'axe atteint le fin de course matériel pendant que le référencement est actif, la prise de référence est interrompue avec une erreur et l'axe est freiné avec la décélération d'urgence.</p>
Sens d'accostage (référencement actif et passif)	Avec la sélection de sens, vous définissez le sens d'accostage utilisé pendant le référencement actif pour rechercher la came de référence, ainsi que le sens de référencement. Le sens de référencement indique le sens de déplacement utilisé par l'axe pour accoster le côté configuré de la came de référence afin d'effectuer le référencement.

Paramètre	Description
Came de référence (référencement actif et passif)	<ul style="list-style-type: none"> Référencement actif : Indiquez si l'axe doit être référencé du côté gauche ou droit de la came de référence. Selon la position de départ de l'axe et la configuration des paramètres de référencement, la séquence de prise de référence peut différer du schéma dans la fenêtre de configuration. Référencement passif : Avec le référencement passif, les mouvements de déplacement aux fins de référencement doivent être réalisés par l'utilisateur à l'aide de commandes de mouvement. Le côté de la came de référence sur lequel se fait le référencement dépend des facteurs suivants : <ul style="list-style-type: none"> Configuration du sens d'accostage Configuration de la came de référence Sens de déplacement actif pendant le référencement passif
Vitesse d'attaque (référencement actif uniquement)	<p>Indiquez la vitesse à laquelle la came de référence est recherchée pendant la prise de référence.</p> <p>Valeurs limites (indépendantes de l'unité utilisateur sélectionnée) : vitesse de démarrage/d'arrêt ≤ vitesse d'attaque ≤ vitesse maximale</p>
Vitesse d'accostage (référencement actif uniquement)	<p>Indiquez la vitesse à laquelle l'axe doit accoster la came de référence pour réaliser le référencement.</p> <p>Valeurs limites (indépendantes de l'unité utilisateur sélectionnée) : vitesse de démarrage/d'arrêt ≤ vitesse d'accostage ≤ vitesse maximale</p>
Décalage de la position de référence (référencement actif uniquement)	<p>Si le point de référence désiré diffère de la position de la came de référence, vous pouvez indiquer le décalage de la position de référence dans ce champ.</p> <p>Si cette valeur est différente de 0, l'axe exécute les actions suivantes après le référencement au niveau de la came de référence :</p> <ol style="list-style-type: none"> Déplacez l'axe de la valeur du décalage de la position de référence à la vitesse d'accostage. Lorsque la position du décalage de la position de référence est atteinte, la position de l'axe prend la valeur de la position de référence absolue. La position de référence absolue est indiquée via le paramètre "Position" de l'instruction de commande de mouvement "MC_Home". <p>Valeurs limites (indépendantes de l'unité utilisateur sélectionnée) : -1,0e12 ≤ décalage de la position de référence ≤ 1,0e12</p>

Tableau 10- 25 Facteurs affectant le référencement

Facteurs déterminants :			Résultat :
Configuration du sens d'accostage	Configuration de la came de référence	Sens de déplacement en cours	Référencement sur la came de référence
Positif	Côté gauche (négatif)	Sens positif	Gauche
		Sens négatif	Droite
Positif	Côté droit (positif)	Sens positif	Droite
		Sens négatif	Gauche
Négatif	Côté gauche (négatif)	Sens positif	Droite
		Sens négatif	Gauche
Négatif	Côté droit (positif)	Sens positif	Gauche
		Sens négatif	Droite

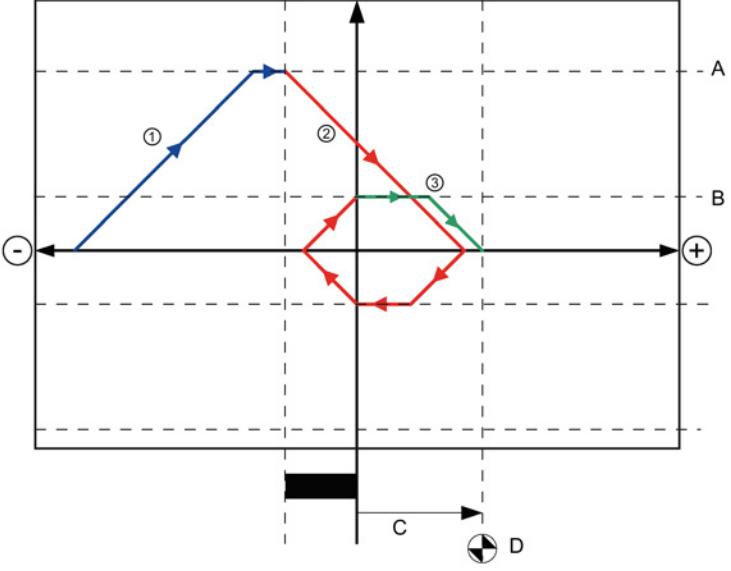
10.6.3.3 Séquence pour le référencement actif

Vous démarrez le référencement actif avec l'instruction de commande de mouvement "MC_Home" (paramètre d'entrée Mode = 3). Dans ce cas, le paramètre d'entrée Position" indique les coordonnées absolues du point de référence. Vous pouvez également lancer le référencement actif dans le panneau de commande à des fins de test.

Le schéma suivant montre un exemple de courbe caractéristique pour une prise de référence active avec les paramètres de configuration suivants :

- sens d'accostage = sens d'accostage positif
- came de référence = côté droit (positif)
- valeur du décalage de la position de référence > 0

Tableau 10- 26 Caractéristiques de vitesse du référencement MC

Fonctionnement		Remarques	
	A	Vitesse d'attaque	
	B	Vitesse d'accostage	
	C	Coordonnée de la position de référence	
	D	Décalage de la position de référence	
①	Phase de recherche (segment bleu de la courbe) : Au démarrage du référencement actif, l'axe accélère jusqu'à la vitesse d'attaque configurée et recherche la came de référence à cette vitesse.		
②	Accostage du point de référence (segment rouge de la courbe) : Une fois la came de référence détectée, l'axe freine et exécute une inversion de sens pour être référencé du côté configuré de la came de référence à la vitesse d'accostage configurée.		
③	Déplacement vers la position du point de référence (segment vers de la courbe) : Après son référencement à la came de référence, l'axe se déplace vers les coordonnées du point de référence à la vitesse d'accostage. Lorsqu'il atteint les coordonnées du point de référence, l'axe est arrêté à la valeur de position indiquée dans le paramètre d'entrée Position de l'instruction MC_Home.		

Remarque

Si la recherche de référencement ne fonctionne pas comme vous le prévoyiez, vérifiez les entrées affectées aux fins de course matériels ou à la came de référence. Les alarmes sur front de ces entrées ont peut-être été désactivées dans la configuration d'appareil.

Examinez les données de configuration de l'objet technologique Axe concerné pour voir, le cas échéant, les entrées qui sont affectées à "HW Low Limit Switch Input", "HW High Limit Switch Input" et "Input reference point switch". Puis, ouvrez la configuration d'appareil pour la CPU et examinez chacune des entrées affectées. Vérifiez que les options "Activer la détection de front montant" et "Activer la détection de front descendant" sont toutes deux sélectionnées. Si elles ne le sont pas, effacez les entrées concernées dans la configuration de l'axe puis sélectionnez-les à nouveau.

10.7 Instructions Motion Control

10.7.1 Présentation des instructions MC

Les instructions de commande de mouvement utilisent un bloc de données technologique associé et les sorties PTO (sorties de trains d'impulsions) réservées de la CPU pour piloter le mouvement sur un axe.

- MC_Power (Page 325) active et désactive un axe de commande de mouvement.
- MC_Reset (Page 328) remet à zéro toutes les erreurs de commande de mouvement. Toutes les erreurs de commande de mouvement pouvant être acquittées le sont.
- MC_Home (Page 329) établit la relation entre le programme de commande d'axe et le système de positionnement mécanique de l'axe.
- MC_Halt (Page 332) annule tous les processus de mouvement et provoque l'arrêt du mouvement de l'axe. La position d'arrêt n'est pas définie.
- MC_MoveAbsolute (Page 334) démarre le mouvement vers une position absolue. Le travail s'achève lorsque la position cible est atteinte.
- MC_MoveRelative (Page 336) démarre un mouvement de positionnement relatif par rapport à la position de départ.
- MC_MoveVelocity (Page 338) provoque le mouvement de l'axe à la vitesse indiquée.
- MC_MoveJog (Page 341) exécute le mode manuel à vue aux fins de test et de mise en route.
- MC_CommandTable (Page 344) exécute des commandes d'axe en tant que séquence de mouvement.
- MC_ChangeDynamic (Page 347) modifie les réglages dynamiques pour l'axe.
- MC_WriteParam (Page 349) écrit dans un nombre choisi de paramètres pour modifier la fonctionnalité de l'axe à partir du programme utilisateur.
- MC_ReadParam (Page 351) lit un nombre choisi de paramètres qui indiquent la position en cours, la vitesse, etc., de l'axe défini dans l'entrée Axis.

Niveaux de firmware de la CPU

Si vous avez une CPU S7-1200 avec le firmware V4.1, sélectionnez la version V5.0 de chaque instruction de mouvement.

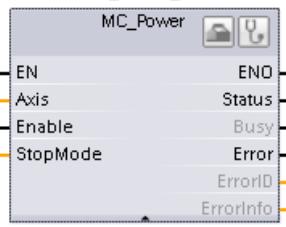
Si vous avez une CPU S7-1200 avec un firmware V4.0 ou antérieur, sélectionnez la version V4.0, V3.0, V2.0 ou V1.0 appropriée de chaque instruction de mouvement.

10.7.2 Instruction MC_Power (Libérer/bloquer l'axe)

Remarque

Si l'axe est désactivé à la suite d'une erreur, il sera automatiquement réactivé une fois l'erreur éliminée et acquittée. Cela nécessite que le paramètre d'entrée Enable ait conservé la valeur VRAI pendant ce processus.

Tableau 10- 27 Instruction MC_Power

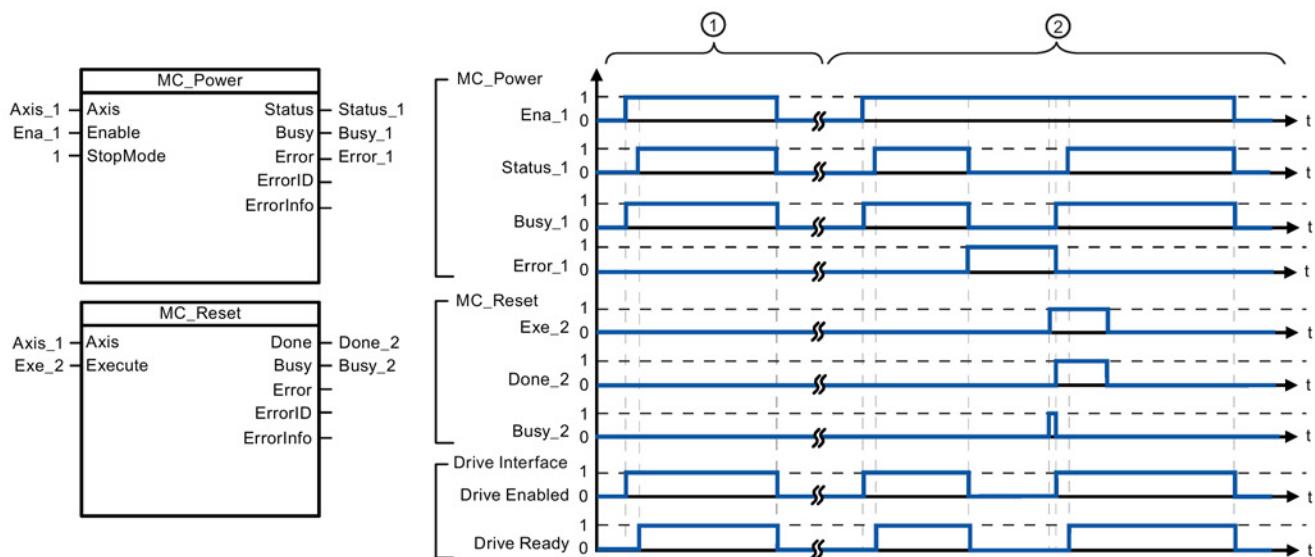
CONT/LOG	SCL	Description
	<pre> "MC_Power_DB"(Axis:=_multi_fb_in_, Enable:=_bool_in_, StopMode:=_int_in_, Status=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorIn- fo=>_word_out_); </pre>	<p>L'instruction de commande de mouvement MC_Power valide ou inhibe un axe. Les conditions suivantes doivent être remplies pour que vous puissiez libérer ou bloquer l'axe :</p> <ul style="list-style-type: none"> • L'objet technologique a été configuré correctement. • Il n'y a pas d'erreur de libération/blocage en attente. <p>L'exécution de MC_Power ne peut pas être annulée par une tâche de commande de mouvement. Le blocage de l'axe (paramètre d'entrée Enable = FAUX) annule toutes les tâches de commande de mouvement pour l'objet technologique associé.</p>

¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

² Dans l'exemple SCL, "MC_Power_DB" est le nom du DB d'instance.

Tableau 10- 28 Paramètres pour l'instruction MC_Power

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_1 Objet technologique Axe
Enable	IN	Bool <ul style="list-style-type: none"> • FALSE (valeur par défaut) : Toutes les tâches actives sont annulées conformément au mode "StopMode" paramétré et l'axe s'arrête. • TRUE : La commande de mouvement tente de valider l'axe.
StopMode	IN	Int <ul style="list-style-type: none"> • 0 : Arrêt d'urgence : Si une demande d'inhibition de l'axe est en cours, l'axe exécute la décélération d'urgence configurée. L'axe est bloqué dès qu'il est immobilisé. • 1 : Arrêt immédiat : Si une demande d'inhibition de l'axe est en cours, l'axe est arrêté sans décélération. La sortie d'impulsions est arrêtée immédiatement. • 2 : Arrêt d'urgence avec contrôle de jerk : Si une demande d'inhibition de l'axe est en cours, l'axe exécute la décélération d'urgence configurée. Si la commande d'à-coup est activée, l'à-coup configuré est pris en compte. L'axe est bloqué dès qu'il est immobilisé.
Status	OUT	Bool Etat de libération de l'axe : <ul style="list-style-type: none"> • FALSE : L'axe est bloqué : <ul style="list-style-type: none"> – L'axe n'exécute pas de tâches de commande de mouvement et n'accepte pas de nouvelles tâches (à l'exception d'une tâche MC_Reset). – L'axe n'est pas référencé. – Lors du blocage, l'état ne passe pas à FAUX tant que l'axe n'est pas immobilisé. • TRUE : L'axe est libéré : <ul style="list-style-type: none"> – L'axe est prêt à exécuter des tâches de commande de mouvement. – Lors de la libération de l'axe, l'état ne passe à VRAI que lorsque le signal "Entraînement prêt" est présent. Si l'interface d'entraînement "Entraînement prêt" n'a pas été configurée dans la configuration de l'axe, l'état passe à VRAI immédiatement.
Busy	OUT	Bool FALSE : MC_Power n'est pas actif. TRUE : MC_Power est actif.
Error	OUT	Bool FALSE : Pas d'erreur TRUE : Une erreur s'est produite dans l'instruction de commande de mouvement "MC_Power" ou dans l'objet technologique associé. La cause de l'erreur figure dans les paramètres "ErrorID" et "ErrorInfo".
ErrorID	OUT	Word Code d'erreur pour le paramètre "Error"
ErrorInfo	OUT	Word Information d'erreur pour le paramètre "ErrorID"



- ① Un axe est validé puis à nouveau bloqué. Une fois que l'entraînement a renvoyé "Entraînement prêt" à la CPU, la réussite de la libération peut être lue via "Status_1".
- ② Après une libération d'axe, une erreur ayant entraîné le blocage de l'axe s'est produite. L'erreur est supprimée et acquittée avec "MC_Reset". L'axe est alors à nouveau libéré.

Procédez comme suit pour libérer un axe à interface d'entraînement configurée :

1. Vérifiez que les conditions décrites plus haut sont bien remplies.
2. Initialisez le paramètre d'entrée "StopMode" à la valeur désirée. Définissez le paramètre d'entrée "Enable" à VRAI.

La sortie de libération pour "Entraînement libéré" passe à VRAI pour laisser passer le courant vers l'entraînement. La CPU attend le signal "Entraînement prêt" de l'entraînement.

Lorsque le signal "Entraînement prêt" est disponible sur l'entrée Prêt configurée de la CPU, l'axe est libéré. Le paramètre de sortie "Status" et la variable d'objet technologique <nom axe>.StatusBits.Enable prennent la valeur VRAI.

Procédez comme suit pour libérer un axe sans interface d'entraînement configurée :

1. Vérifiez que les conditions décrites plus haut sont bien remplies.
2. Initialisez le paramètre d'entrée "StopMode" à la valeur désirée. Définissez le paramètre d'entrée "Enable" à VRAI. L'axe est libéré. Le paramètre de sortie "Status" et la variable d'objet technologique <nom d'axe>.StatusBits.Enable prennent la valeur VRAI.

Procédez comme suit pour bloquer un axe :

1. Immobilisez l'axe.
2. Vous pouvez savoir quand l'axe est immobile à l'aide de la variable d'objet technologique <nom d'axe>.StatusBits.StandStill.
3. Définissez le paramètre d'entrée "Enable" à FAUX une fois l'axe immobilisé.
4. Les paramètres de sortie "Busy" et "Status" et la variable d'objet technologique <nom d'axe>.StatusBits.Enable prennent la valeur FAUX lorsque le blocage de l'axe est achevé.

10.7.3 Instruction MC_Reset (Confirmer l'erreur)

Tableau 10- 29 Instruction MC_Reset

CONT/LOG	SCL	Description
	<pre>"MC_Reset_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Restart:=_bool_in_, Done=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilisez l'instruction MC_Reset pour acquitter les erreurs "Erreur de fonctionnement avec arrêt de l'axe" et "Erreur de configuration". Vous trouverez les erreurs nécessitant un acquittement dans la liste des ErrorIDs et ErrorInfos, sous "Solution".</p> <p>Avant d'utiliser l'instruction MC_Reset, vous devez avoir éliminé la cause de l'erreur de configuration en attente nécessitant un acquittement (par exemple, en changeant une valeur d'accélération invalide dans l'objet technologique Axe en une valeur valide).</p> <p>A partir de V3.0, le paramètre Restart permet de charger la configuration d'axe en mémoire de travail à l'état de fonctionnement MARCHE.</p>

¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

² Dans l'exemple SCL, "MC_Reset_DB" est le nom du DB d'instance.

La tâche MC_Reset ne peut être annulée par aucune autre tâche de commande de mouvement. La nouvelle tâche MC_Reset n'annule aucune autre tâche de commande de mouvement active.

Tableau 10- 30 Paramètres pour l'instruction MC_Reset

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_1 Objet technologique Axe
Execute	IN	Bool Démarrage de la tâche en présence d'un front montant
Restart	IN	TRUE = Chargement de la configuration d'axe de la mémoire de chargement dans la mémoire de travail. La commande ne peut être exécutée que lorsque l'axe est bloqué. FALSE = Acquittement des erreurs en attente
Done	OUT	VRAl = L'erreur a été acquittée.
Busy	OUT	VRAl = La tâche est en cours d'exécution.
Error	OUT	VRAl = Une erreur s'est produite pendant l'exécution de la tâche. La cause de l'erreur figure dans les paramètres "ErrorID" et "ErrorInfo".
ErrorID	OUT	Code d'erreur pour le paramètre "Error"
ErrorInfo	OUT	Information d'erreur pour le paramètre "ErrorID"

Procédez comme suit pour acquitter une erreur avec MC_Reset :

1. Vérifiez que les conditions décrites plus haut sont bien remplies.
2. Lancez l'acquittement de l'erreur en présence d'un front montant dans le paramètre d'entrée Execute.
3. On sait que l'erreur a été acquittée lorsque Done est égal à VRAl et que la variable d'objet technologique <nom d'axe>.StatusBits.Error est égal à FAUX.

10.7.4 Instruction MC_Home (Référencer l'axe)

Tableau 10- 31 Instruction MC_Home

CONT/LOG	SCL	Description
	<pre>"MC_Home_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Position:=_real_in_, Mode:=_int_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_); </pre>	<p>Utilisez l'instruction MC_Home pour établir une correspondance entre les coordonnées de l'axe et la position physique réelle de l'entraînement. Le référencement est nécessaire pour un positionnement absolu de l'axe.</p> <p>L'utilisation de l'instruction MC_Home n'est possible que si l'axe est libéré</p>

¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

² Dans l'exemple SCL, "MC_Home_DB" est le nom du DB d'instance.

Les modes de référencement suivants sont disponibles :

- Référencement direct absolu (Mode = 0) : La valeur du paramètre "Position" est prise comme position en cours de l'axe.
- Référencement direct relatif (Mode = 1) : La position en cours de l'axe est décalée de la valeur du paramètre "Position".
- Référencement passif (Mode = 2) : Lors du référencement passif, l'instruction MC_Home n'exécute aucune prise de référence. L'accostage de la came de référence doit être réalisé par l'utilisateur au moyen d'instructions de commande de mouvement. L'axe est référencé lorsque la came de référence est détectée.
- Référencement actif (Mode = 3) : La prise de référence est exécutée automatiquement.

Tableau 10- 32 Paramètres pour l'instruction MC_Home

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_PTO Objet technologique Axe
Execute	IN	Bool Démarrage de la tâche en présence d'un front montant
Position	IN	Real <ul style="list-style-type: none"> Mode = 0, 2 et 3 (position absolue de l'axe à l'achèvement du référencement) Mode = 1 (valeur de correction pour la position en cours de l'axe) Valeurs limites : $-1,0e^{12} \leq \text{Position} \leq 1,0e^{12}$
Mode	IN	Int Mode de référencement <ul style="list-style-type: none"> 0 : Référencement direct absolu La nouvelle position de l'axe correspond à la valeur de position du paramètre "Position". 1 : Référencement direct relatif La nouvelle position de l'axe correspond à la position en cours de l'axe + la valeur de position du paramètre "Position". 2 : Référencement passif Référencement selon la configuration de l'axe. Après le référencement, la valeur du paramètre "Position" est prise comme nouvelle position de l'axe. 3 : Référencement actif Prise de référence selon la configuration de l'axe. Après le référencement, la valeur du paramètre "Position" est prise comme nouvelle position de l'axe.
Done	OUT	Bool VRAI = Tâche achevée
Busy	OUT	Bool VRAI = La tâche est en cours d'exécution.
CommandAborted	OUT	Bool VRAI = Pendant l'exécution, la tâche a été annulée par une autre tâche.
Error	OUT	Bool VRAI = Une erreur s'est produite pendant l'exécution de la tâche. La cause de l'erreur figure dans les paramètres "ErrorID" et "ErrorInfo".
ErrorID	OUT	Word Code d'erreur pour le paramètre "Error"
ErrorInfo	OUT	Word Information d'erreur pour le paramètre "ErrorID"

Remarque**Perte du référencement de l'axe dans les cas suivants**

- Blocage de l'axe par l'instruction MC_Power
- Commutation entre mode automatique et mode manuel
- Lors du démarrage d'un référencement actif (le référencement de l'axe est à nouveau disponible à l'achèvement de l'opération de référencement)
- Après mise hors tension puis sous tension de la CPU
- Après un redémarrage de la CPU (transition MARCHE à ARRET ou ARRET à MARCHE)

Procédez comme suit pour référencer l'axe :

1. Vérifiez que les conditions décrites plus haut sont bien remplies.
2. Initialisez les paramètres d'entrée nécessaires et lancez l'opération de référencement à l'aide d'un front montant dans le paramètre d'entrée "Execute".
3. On sait que le référencement est achevé lorsque le paramètre de sortie "Done" et la variable d'objet technologique <nom d'axe>.StatusBits.HomingDone prennent la valeur VRAI.

Tableau 10- 33 Comportement d'annulation

Mode	Description	
0 ou 1	La tâche MC_Home ne peut être annulée par aucune autre tâche de commande de mouvement. La nouvelle tâche MC_Home n'annule aucune tâche de commande de mouvement active. Les tâches de déplacement par rapport à une position reprennent après le référencement en fonction de la nouvelle position de référence (valeur dans le paramètre d'entrée Position).	
2	La tâche MC_Home peut être annulée par les tâches de commande de mouvement suivantes : Tâche MC_Home Mode = 2, 3 : La nouvelle tâche MC_Home annule la tâche de commande de mouvement active suivante. Tâche MC_Home Mode = 2 : Les tâches de déplacement par rapport à une position reprennent après le référencement en fonction de la nouvelle position de référence (valeur dans le paramètre d'entrée Position).	
3	<p>La tâche MC_Home peut être annulée par les tâches de commande de mouvement suivantes :</p> <ul style="list-style-type: none"> • MC_Home Mode = 3 • MC_Halt • MC_MoveAbsolute • MC_MoveRelative • MC_MoveVelocity • MC_MoveJog 	<p>La nouvelle tâche MC_Home annule les tâches de commande de mouvement actives suivantes :</p> <ul style="list-style-type: none"> • MC_Home Mode = 2, 3 • MC_Halt • MC_MoveAbsolute • MC_MoveRelative • MC_MoveVelocity • MC_MoveJog

10.7.5 Instruction MC_Halt (Pause de l'axe)

Tableau 10- 34 Instruction MC_Halt

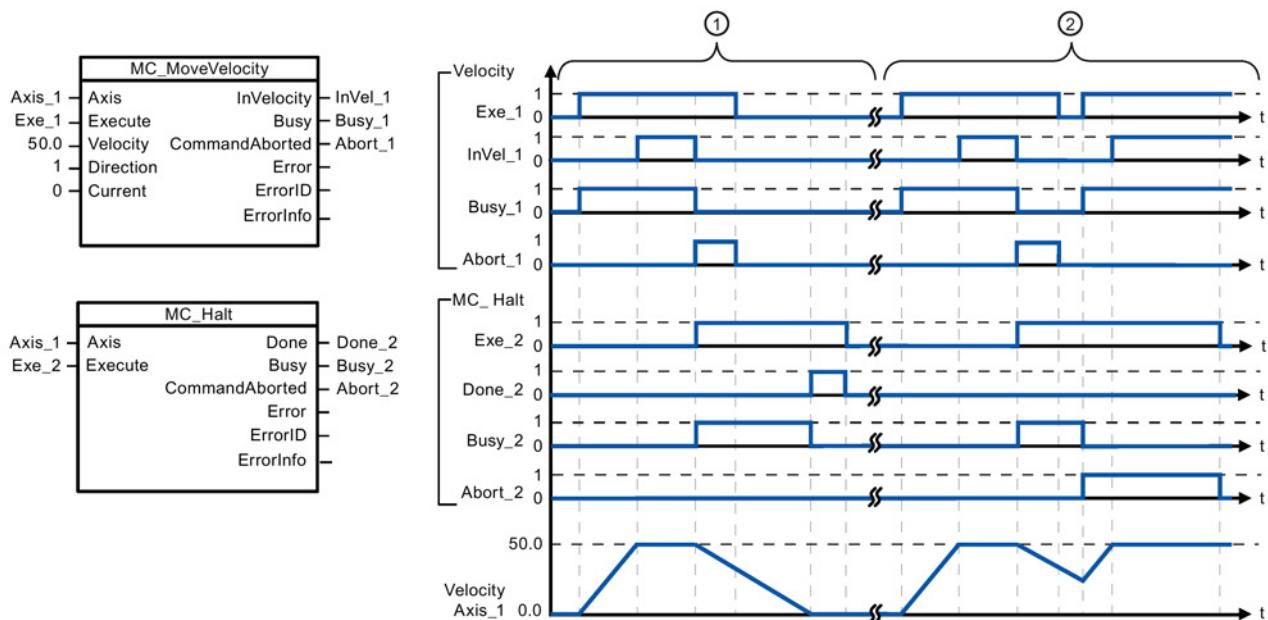
CONT/LOG	SCL	Description
 <p>"MC_Halt_DB"</p> <p>MC_Halt</p> <p>EN</p> <p>Axis</p> <p>Execute</p> <p>ENO</p> <p>Done</p> <p>Busy</p> <p>CommandAborted</p> <p>Error</p> <p>ErrorCode</p> <p>ErrorInfo</p>	<pre>"MC_Halt_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilisez l'instruction MC_Halt pour arrêter tout mouvement et immobiliser l'axe. La position d'immobilisation n'est pas définie. L'utilisation de l'instruction MC_Halt n'est possible que si l'axe est libéré.</p>

1 STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

2 Dans l'exemple SCL, "MC_Halt_DB" est le nom du DB d'instance.

Tableau 10- 35 Paramètres pour l'instruction MC_Halt

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_1
Execute	IN	Bool
Done	OUT	Bool
Busy	OUT	Bool
CommandAborted	OUT	Bool
Error	OUT	Bool
ErrorID	OUT	Word
ErrorInfo	OUT	Word



Les valeurs suivantes ont été configurées dans la fenêtre de configuration "Dynamique > Générale" : Accélération = 10,0 et Décélération = 5,0

- ① L'axe est freiné jusqu'à l'immobilisation par une tâche MC_Halt. L'immobilisation de l'axe est signalée via "Done_2".
- ② Alors qu'une tâche MC_Halt freine l'axe, cette tâche est annulée par une autre tâche de mouvement. L'annulation est signalée via "Abort_2".

Comportement d'annulation

La tâche MC_Halt peut être annulée par les tâches de commande de mouvement suivantes :

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

La nouvelle tâche MC_Halt annule les tâches de commande de mouvement actives suivantes :

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

10.7.6 Instruction MC_MoveAbsolute (Positionnement absolu de l'axe)

Tableau 10- 36 Instruction MC_MoveAbsolute

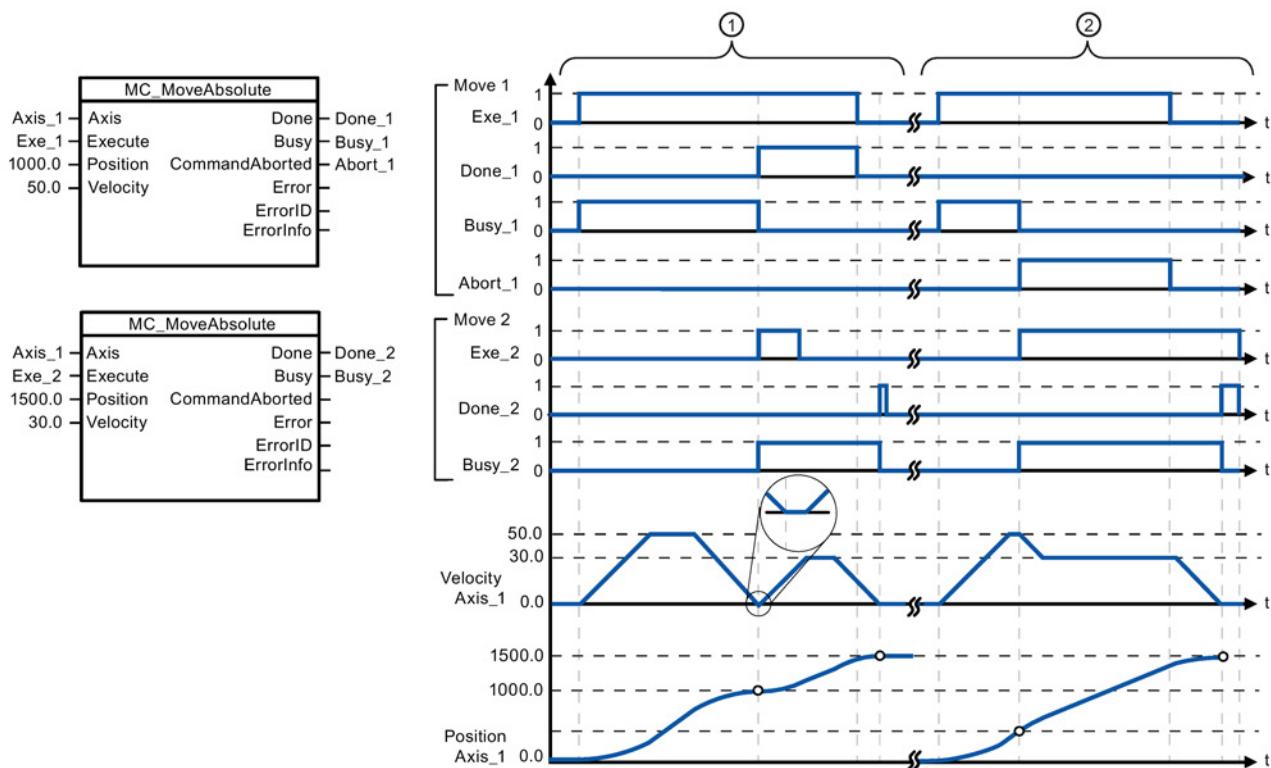
CONT/LOG	SCL	Description
	<pre>"MC_MoveAbsolute_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Position:=_real_in_, Velocity:=_real_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilisez l'instruction MC_MoveAbsolute pour démarrer un déplacement de positionnement de l'axe à une position absolue.</p> <p>L'utilisation de l'instruction MC_MoveAbsolute n'est possible que si l'axe est libéré et référencé.</p>

¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

² Dans l'exemple SCL, "MC_MoveAbsolute_DB" est le nom du DB d'instance.

Tableau 10- 37 Paramètres pour l'instruction MC_MoveAbsolute

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_1 Objet technologique Axe
Execute	IN	Bool Démarrage de la tâche en présence d'un front montant (valeur par défaut : faux)
Position	IN	Real Position cible absolue (valeur par défaut : 0,0) 0,0 Valeurs limites : -1,0e12 ≤ Position ≤ 1,0e12
Velocity	IN	Real Vitesse de l'axe (valeur par défaut : 10,0) Cette vitesse n'est pas toujours atteinte en raison de l'accélération et de la décélération configurées ainsi que de la position cible à accoster. Valeurs limites : vitesse de démarrage/d'arrêt ≤ Velocity ≤ vitesse maximale
Done	OUT	Bool VRAI = La position cible absolue est atteinte.
Busy	OUT	Bool VRAI = La tâche est en cours d'exécution.
CommandAborted	OUT	Bool VRAI = Pendant l'exécution, la tâche a été annulée par une autre tâche.
Error	OUT	Bool VRAI = Une erreur s'est produite pendant l'exécution de la tâche. La cause de l'erreur figure dans les paramètres "ErrorID" et "ErrorInfo".
ErrorID	OUT	Word Code d'erreur pour le paramètre "Error" (valeur par défaut : 0000) 0000)
ErrorInfo	OUT	Word Information d'erreur pour le paramètre "ErrorID" (valeur par défaut : 0000) 0000)



Les valeurs suivantes ont été configurées dans la fenêtre de configuration "Dynamique > Générale" : Accélération = 10,0 et Décélération = 10,0

- ① Un axe est déplacé à la position absolue 1000,0 avec une tâche `MC_MoveAbsolute`. Lorsque l'axe atteint la position cible, cela est signalé via "Done_1". Lorsque "Done_1" = VRAI, une autre tâche `MC_MoveAbsolute`, avec une position cible 1500,0, est lancée. Du fait des temps de réponse (par exemple, temps de cycle du programme utilisateur, etc.), l'axe s'immobilise brièvement (voir le détail agrandi). Lorsque l'axe atteint la nouvelle position cible, cela est signalé via "Done_2".
- ② Une tâche `MC_MoveAbsolute` active est annulée par une autre tâche `MC_MoveAbsolute`. L'annulation est signalée via "Abort_1". L'axe est ensuite déplacé à la nouvelle vitesse vers la nouvelle position cible 1500,0. Lorsque la nouvelle position cible est atteinte, cela est signalé via "Done_2".

Comportement d'annulation

La tâche `MC_MoveAbsolute` peut être annulée par les tâches de commande de mouvement suivantes :

- `MC_Home Mode` = 3
- `MC_Halt`
- `MC_MoveAbsolute`
- `MC_MoveRelative`
- `MC_MoveVelocity`
- `MC_MoveJog`

La nouvelle tâche `MC_MoveAbsolute` annule les tâches de commande de mouvement actives suivantes :

- `MC_Home Mode` = 3
- `MC_Halt`
- `MC_MoveAbsolute`
- `MC_MoveRelative`
- `MC_MoveVelocity`
- `MC_MoveJog`

10.7.7 Instruction MC_MoveRelative (Positionnement relatif de l'axe)

Tableau 10- 38 Instruction MC_MoveRelative

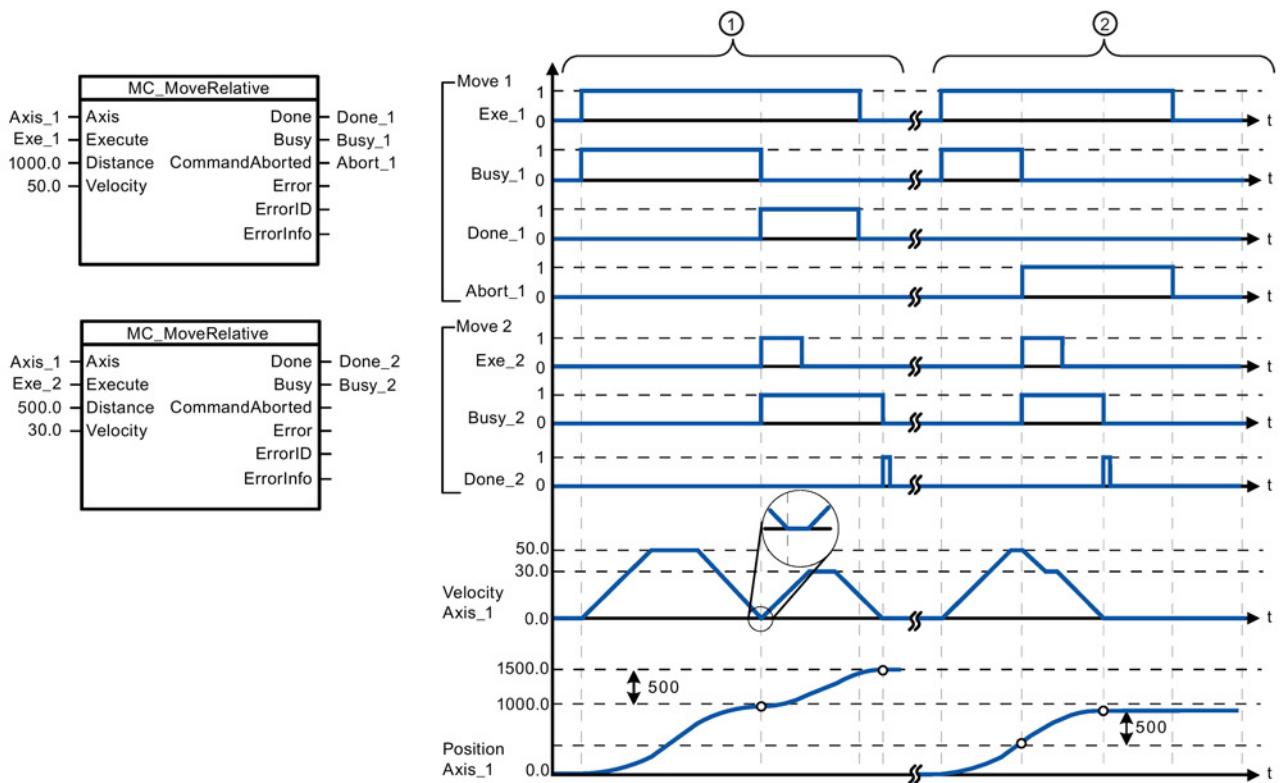
CONT/LOG	SCL	Description
	<pre>"MC_MoveRelative_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Distance:=_real_in_, Velocity:=_real_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilisez l'instruction MC_MoveRelative pour démarrer un déplacement de positionnement relatif de l'axe par rapport à une position initiale..</p> <p>L'utilisation de l'instruction MC_MoveRelative n'est possible que si l'axe est libéré.</p>

¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

² Dans l'exemple SCL, "MC_MoveRelative_DB" est le nom du DB d'instance.

Tableau 10- 39 Paramètres pour l'instruction MC_MoveRelative

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_1 Objet technologique Axe
Execute	IN	Bool Démarrage de la tâche en présence d'un front montant (valeur par défaut : faux)
Distance	IN	Real Distance à parcourir pour l'opération de positionnement (valeur par défaut : 0,0) 0,0) Valeurs limites : -1,0e ¹² ≤ Distance ≤ 1,0e ¹²
Velocity	IN	Real Vitesse de l'axe (valeur par défaut : 10,0) Cette vitesse n'est pas toujours atteinte en raison de l'accélération et de la décélération configurées ainsi que de la distance à parcourir. Valeurs limites : vitesse de démarrage/d'arrêt ≤ Velocity ≤ vitesse maximale
Done	OUT	Bool VRAI = La position cible est atteinte.
Busy	OUT	Bool VRAI = La tâche est en cours d'exécution.
CommandAborted	OUT	Bool VRAI = Pendant l'exécution, la tâche a été annulée par une autre tâche.
Error	OUT	Bool VRAI = Une erreur s'est produite pendant l'exécution de la tâche. La cause de l'erreur figure dans les paramètres "ErrorID" et "ErrorInfo".
ErrorID	OUT	Word Code d'erreur pour le paramètre "Error" (valeur par défaut : 0000 0000)
ErrorInfo	OUT	Word Information d'erreur pour le paramètre "ErrorID" (valeur par défaut : 0000 0000)



Les valeurs suivantes ont été configurées dans la fenêtre de configuration "Dynamique > Générale" : Accélération = 10,0 et Décélération = 10,0

- ① L'axe est déplacé par une tâche `MC_MoveRelative` de la distance ("Distance") 1000,0. Lorsque l'axe atteint la position cible, cela est signalé via "Done_1". Lorsque "Done_1" = VRAI, une autre tâche `MC_MoveRelative`, avec une distance de déplacement de 500,0, est lancée. Du fait des temps de réponse (par exemple, temps de cycle du programme utilisateur), l'axe s'immobilise brièvement (voir le détail agrandi). Lorsque l'axe atteint la nouvelle position cible, cela est signalé via "Done_2".
- ② Une tâche `MC_MoveRelative` active est annulée par une autre tâche `MC_MoveRelative`. L'annulation est signalée via "Abort_1". L'axe est ensuite déplacé de la nouvelle distance ("Distance") 500,0 à la nouvelle vitesse. Lorsque la nouvelle position cible est atteinte, cela est signalé via "Done_2".

Comportement d'annulation

La tâche `MC_MoveRelative` peut être annulée par les tâches de commande de mouvement suivantes :

- `MC_Home Mode` = 3
- `MC_Halt`
- `MC_MoveAbsolute`
- `MC_MoveRelative`
- `MC_MoveVelocity`
- `MC_MoveJog`

La nouvelle tâche `MC_MoveRelative` annule les tâches de commande de mouvement actives suivantes :

- `MC_Home Mode` = 3
- `MC_Halt`
- `MC_MoveAbsolute`
- `MC_MoveRelative`
- `MC_MoveVelocity`
- `MC_MoveJog`

10.7.8 Instruction MC_MoveVelocity (Déplacer l'axe à une vitesse prédéfinie)

Tableau 10- 40 Instruction MC_MoveVelocity

CONT/LOG	SCL	Description
 <p>"MC_MoveVelocity_DB" (</p> <ul style="list-style-type: none"> Axis:= _multi_fb_in_, Execute:= _bool_in_, Velocity:= _real_in_, Direction:= _int_in_, Current:= _bool_in_, InVelocity=> _bool_out_, Busy=> _bool_out_, CommandAborted=> _bool_out_, Error=> _bool_out_, ErrorID=> _word_out_, ErrorInfo=> _word_out_); 	<p>Utilisez l'instruction MC_MoveVelocity pour déplacer l'axe à la vitesse prédéfinie.</p> <p>L'utilisation de l'instruction MC_MoveVelocity n'est possible que si l'axe est libéré.</p>	

1 STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

2 Dans l'exemple SCL, "MC_MoveVelocity_DB" est le nom du DB d'instance.

Tableau 10- 41 Paramètres pour l'instruction MC_MoveVelocity

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_1
Execute	IN	Bool
Velocity	IN	Real
Direction	IN	Int

Objet technologique Axe

Démarrage de la tâche en présence d'un front montant (valeur par défaut : faux)

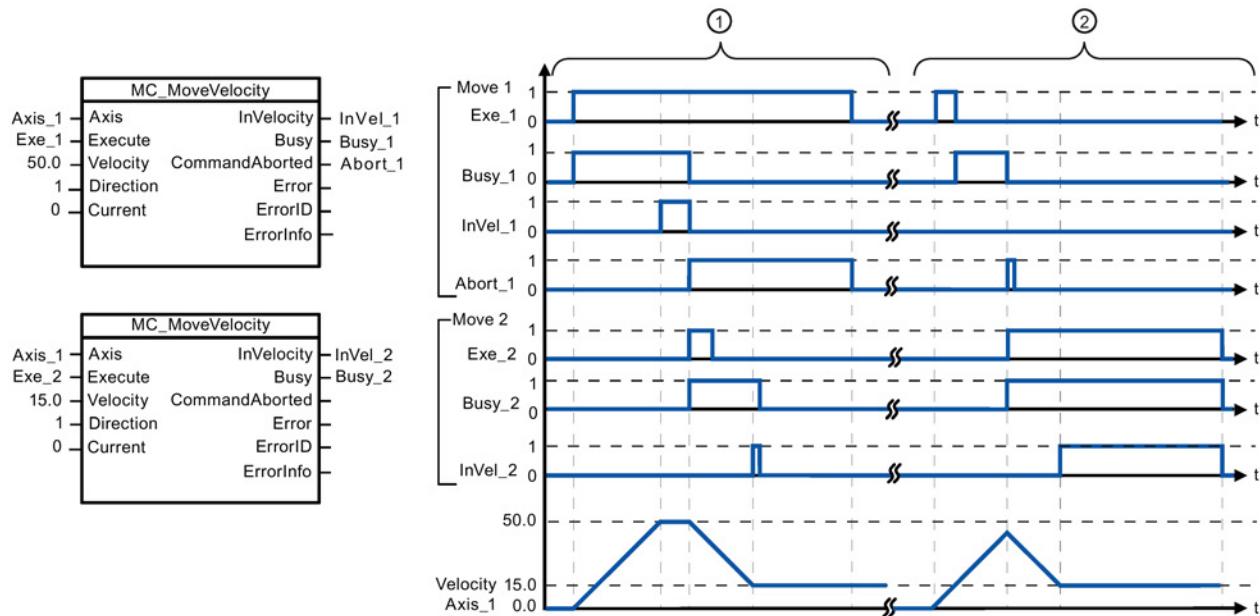
Indication de vitesse pour le déplacement de l'axe (valeur par défaut : 10,0) 10,0

Valeurs limites : vitesse de démarrage/d'arrêt $\leq |Velocity| \leq$ vitesse maximale
(Velocity = 0,0 est autorisé)

Indication du sens :

- 0 : 0 : Le sens de rotation correspond au signe de la valeur dans le paramètre "Velocity" (valeur par défaut).
- 1 : 1 : Sens de rotation positif (le signe de la valeur dans le paramètre "Velocity" n'est pas pris en compte)
- 2 : 2 : Sens de rotation négatif (le signe de la valeur dans le paramètre "Velocity" n'est pas pris en compte)

Paramètre et type	Type de données	Description
Current	IN	Bool Conserver la vitesse actuelle : <ul style="list-style-type: none"> FAUX : "Conserver la vitesse actuelle" est désactivé. Les valeurs des paramètres "Velocity" et "Direction" sont utilisées. (valeur par défaut) VRAI : "Conserver la vitesse actuelle" est activé. Les valeurs des paramètres "Velocity" et "Direction" ne sont pas prises en compte. <p>Lorsque l'axe reprend le déplacement à la vitesse en cours, le paramètre "InVelocity" renvoie la valeur TRUE.</p>
InVelocity	OUT	Bool VRAI : <ul style="list-style-type: none"> Si "Current" = FAUX (FALSE) : La vitesse indiquée dans le paramètre "Velocity" a été atteinte. Si "Current" = VRAI (TRUE) : L'axe se déplace à la vitesse en cours au démarrage.
Busy	OUT	Bool VRAI = La tâche est en cours d'exécution.
CommandAborted	OUT	Bool VRAI = Pendant l'exécution, la tâche a été annulée par une autre tâche.
Error	OUT	Bool VRAI = Une erreur s'est produite pendant l'exécution de la tâche. La cause de l'erreur figure dans les paramètres "ErrorID" et "ErrorInfo".
ErrorID	OUT	Word Code d'erreur pour le paramètre "Error" (valeur par défaut : 0000 0000)
ErrorInfo	OUT	Word Information d'erreur pour le paramètre "ErrorID" (valeur par défaut : 0000 0000)



Les valeurs suivantes ont été configurées dans la fenêtre de configuration "Dynamique > Générale" : Accélération = 10,0 et Décélération = 10,0

- ① Une tâche MC_MoveVelocity active utilise "InVel_1" pour signaler que sa vitesse cible a été atteinte. Elle est ensuite annulée par une autre tâche MC_MoveVelocity. L'annulation est signalée via "Abort_1". Lorsque la nouvelle vitesse cible de 15,0 est atteinte, cela est signalé via "InVel_2". L'axe poursuit alors son déplacement à la nouvelle vitesse constante.
- ② Une tâche active MC_MoveVelocity est annulée par une autre tâche MC_MoveVelocity avant d'atteindre sa vitesse cible. L'annulation est signalée via "Abort_1". Lorsque la nouvelle vitesse cible de 15,0 est atteinte, cela est signalé via "InVel_2". L'axe poursuit alors son déplacement à la nouvelle vitesse constante.

Comportement d'annulation

La tâche MC_MoveVelocity peut être annulée par les tâches de commande de mouvement suivantes :

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

La nouvelle tâche MC_MoveVelocity annule les tâches de commande de mouvement actives suivantes :

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

Remarque**Comportement en cas de vitesse définie à 0 (Velocity = 0,0)**

Une tâche MC_MoveVelocity avec "Velocity" = 0,0 (par exemple, une tâche MC_Halt) annule les tâches de déplacement actives et arrête l'axe avec la décélération configurée. Lorsque l'axe s'immobilise, le paramètre de sortie "InVelocity" indique VRAI pendant au moins un cycle du programme.

"Busy" indique la valeur VRAI pendant l'opération de décélération et passe à FAUX conjointement avec "InVelocity". Si le paramètre "Execute" est VRAI, "InVelocity" et "Busy" sont verrouillés.

Lorsque la tâche MC_MoveVelocity est démarrée, le bit d'état "SpeedCommand" est mis à 1 dans l'objet technologique. Le bit d'état "ConstantVelocity" est mis à 1 lors de l'immobilisation de l'axe. Les deux bits sont adaptés à la nouvelle situation lorsqu'une nouvelle tâche de déplacement est démarrée.

10.7.9 Instruction MC_MoveJog (Déplacer l'axe en mode Manuel à vue)

Tableau 10- 42 Instruction MC_MoveJog

CONT/LOG	SCL	Description
	<pre>"MC_MoveJog_DB" (Axis:= multi_fb_in_, JogForward:= _bool_in_, JogBackward:= _bool_in_, Velocity:= _real_in_, InVelocity=> _bool_out_, Busy=> _bool_out_, CommandAborted=> _bool_out_, Error=> _bool_out_, ErrorID=> _word_out_, ErrorInfo=> _word_out_);</pre>	<p>Utilisez l'instruction MC_MoveJog pour déplacer l'axe à la vitesse prédéfinie en mode Manuel à vue. Cette instruction sert typiquement au test et à la mise en service.</p> <p>L'utilisation de l'instruction MC_MoveJog n'est possible que si l'axe est libéré.</p>

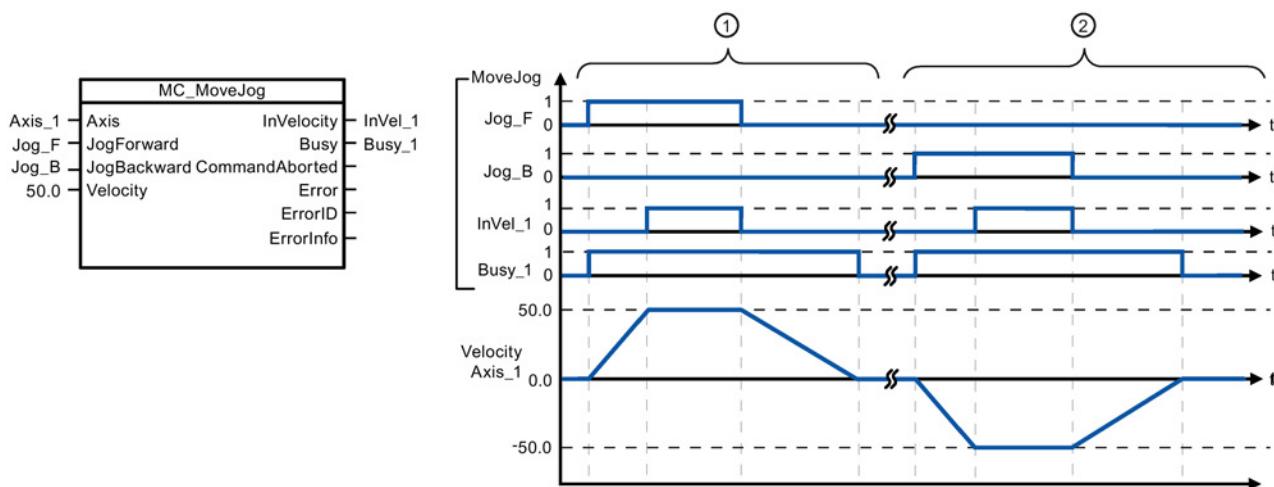
¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

² Dans l'exemple SCL, "MC_MoveJog_DB" est le nom du DB d'instance.

Tableau 10- 43 Paramètres pour l'instruction MC_MoveJog

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_1 Objet technologique Axe
JogForward ¹	IN	Bool Tant que le paramètre est VRAI, l'axe se déplace dans le sens positif à la vitesse indiquée dans le paramètre "Velocity". Le signe de la valeur dans le paramètre "Velocity" n'est pas pris en compte. (valeur par défaut : faux)
JogBackward ¹	IN	Bool Tant que le paramètre est VRAI, l'axe se déplace dans le sens négatif à la vitesse indiquée dans le paramètre "Velocity". Le signe de la valeur dans le paramètre "Velocity" n'est pas pris en compte. (valeur par défaut : faux)
Velocity	IN	Real Vitesse prédéfinie pour le mode Manuel à vue (valeur par défaut : 10,0) 10,0) Valeurs limites : vitesse de démarrage/d'arrêt $\leq Velocity \leq$ vitesse maximale
InVelocity	OUT	Bool VRAI = La vitesse indiquée dans le paramètre "Velocity" a été atteinte.
Busy	OUT	Bool VRAI = La tâche est en cours d'exécution.
CommandAborted	OUT	Bool VRAI = Pendant l'exécution, la tâche a été annulée par une autre tâche.
Error	OUT	Bool VRAI = Une erreur s'est produite pendant l'exécution de la tâche. La cause de l'erreur figure dans les paramètres "ErrorID" et "ErrorInfo".
ErrorID	OUT	Word Code d'erreur pour le paramètre "Error" (valeur par défaut : 0000 0000)
ErrorInfo	OUT	Word Information d'erreur pour le paramètre "ErrorID" (valeur par défaut : 0000 0000)

¹ Si les deux paramètres JogForward et JogBackward sont simultanément à VRAI, l'axe s'arrête avec la décélération configurée. Une erreur est signalée dans les paramètres "Error", "ErrorID" et "ErrorInfo".



Les valeurs suivantes ont été configurées dans la fenêtre de configuration "Dynamique > Générale" : Accélération = 10,0 et Décélération = 5,0

- ① "Jog_F" déplace l'axe dans le sens positif en mode Manuel à vue. Lorsque la vitesse cible de 50,0 est atteinte, cela est signalé via "InVelo_1". L'axe freine jusqu'à s'immobiliser à nouveau après la réinitialisation de Jog_F.
- ② "Jog_B" déplace l'axe dans le sens négatif en mode Manuel à vue. Lorsque la vitesse cible de 50,0 est atteinte, cela est signalé via "InVelo_1". L'axe freine jusqu'à s'immobiliser à nouveau après la réinitialisation de Jog_B.

Comportement d'annulation

La tâche MC_MoveJog peut être annulée par les tâches de commande de mouvement suivantes :

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

La nouvelle tâche MC_MoveJog annule les tâches de commande de mouvement actives suivantes :

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

10.7.10 Instruction MC_CommandTable (Exécuter des commandes d'axe en tant que séquence de mouvement)

Tableau 10- 44 Instruction MC_CommandTable

CONT/LOG	SCL	Description
	<pre>"MC_CommandTable_DB"(Axis:=_multi_fb_in_, CommandTable:=_multi_fb_in_, Execute:=_bool_in_, StartIndex:=_uint_in_, EndIndex:=_uint_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_, CurrentIndex=>_uint_out_, Code=>_word_out_);</pre>	<p>Exécute une série de mouvements individuels pour un axe de commande de mouvement, ces mouvements individuels pouvant être combinés en une séquence de mouvement.</p> <p>Les mouvements individuels sont configurés dans une table de commande d'objet technologique pour la sortie de trains d'impulsions (TO_CommandTable_PTO).</p>

¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

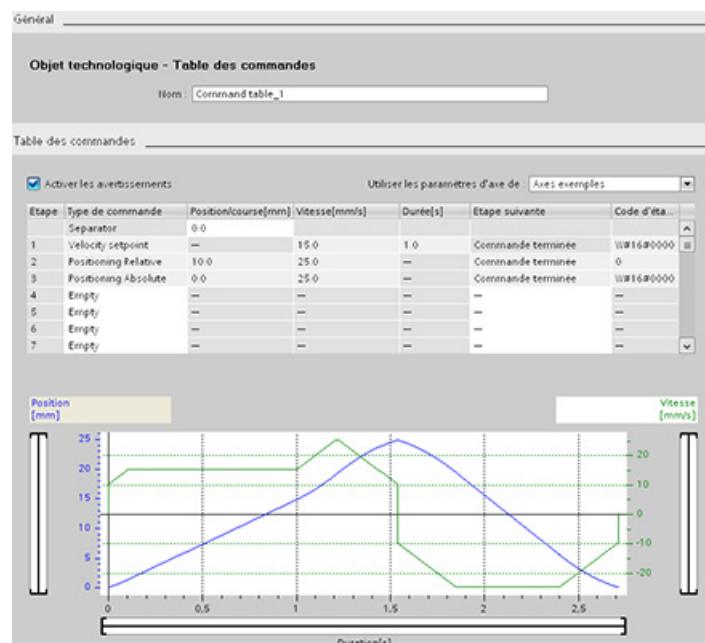
² Dans l'exemple SCL, "MC_CommandTable_DB" est le nom du DB d'instance.

Tableau 10- 45 Paramètres pour l'instruction MC_CommandTable

Paramètre et type	Type de données	Valeur initiale	Description
Axis	IN	TO_Axis_1	- Objet technologique Axe
Table	IN	TO_CommandTable_1	- Objet technologique Table de commande
Execute	IN	Bool	FAUX Démarrage de la tâche en présence d'un front montant
StartIndex	IN	Int	1 Démarrage du traitement de la table de commande à cet indice Limites : 1 ≤ StartIndex ≤ EndIndex
EndIndex	IN	Int	32 Fin du traitement de la table de commande à cet indice Limites : StartIndex ≤ EndIndex ≤ 32
Done	OUT	Bool	FAUX Le traitement de MC_CommandTable a été achevé avec succès.
Busy	OUT	Bool	FAUX Opération en cours
CommandAborted	OUT	Bool	FAUX Le traitement de la tâche a été interrompu par une autre tâche.
Error	OUT	Bool	FAUX Une erreur s'est produite pendant le traitement. La cause est indiquée par les paramètres ErrorID et ErrorInfo..
ErrorID	OUT	Word	16#0000 Code d'erreur
ErrorInfo	OUT	Word	16#0000 Informations d'erreur

Paramètre et type		Type de données	Valeur initiale	Description
Step	OUT	Int	0	Indice en cours de traitement
Code	OUT	Word	16#0000	Identificateur défini par l'utilisateur de l'indice en cours de traitement

Vous pouvez créer la séquence de mouvement désirée dans la fenêtre de configuration "Table de commande" et vérifier le résultat dans la vue graphique du graphique de tendance.



Vous pouvez sélectionner les types de commandes qui doivent être utilisés pour le traitement de la table de commande. Il est possible d'entrer jusqu'à 32 tâches. Les commandes sont traitées l'une après l'autre.

Tableau 10- 46 Types de commandes MC_CommandTable

Type de commande	Description
Empty	Marque de réservation pour toute commande à ajouter. L'entrée Empty n'est pas prise en compte lors du traitement de la table de commande.
Halt	Pause de l'axe. Remarque : cette commande n'est exécutée qu'après une commande de consigne de vitesse.
Positioning Relative	Positionnement de l'axe en fonction de la distance. La commande déplace l'axe de la distance indiquée à la vitesse donnée.
Positioning Absolute	Positionnement de l'axe en fonction de l'emplacement. La commande déplace l'axe à l'emplacement indiqué, en utilisant la vitesse donnée.
Velocity setpoint	Déplacement de l'axe à la vitesse donnée.
Wait	Attente de l'expiration de la durée indiquée. "Wait" n'interrompt pas un accostage actif.
Separator	Ajout d'une ligne séparatrice au-dessus de la ligne sélectionnée. La ligne séparatrice permet de définir plus d'un profil dans une seule table de commande.

Conditions requises pour l'exécution de MC_CommandTable :

- L'objet technologique TO_Axis_PTO V2.0 doit être correctement configuré.
- L'objet technologique TO_CommandTable_PTO doit être correctement configuré.
- L'axe doit être débloqué.

Comportement d'annulation

La tâche MC_CommandTable peut être annulée par les tâches de commande de mouvement suivantes :

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog
- MC_CommandTable

La nouvelle tâche MC_CommandTable annule les tâches de commande de mouvement actives suivantes :

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog
- MC_CommandTable
- La tâche de commande de mouvement en cours avec le lancement de la première commande "Positioning Relative", "Positioning Absolute", "Velocity setpoint" ou "Halt"

10.7.11 Instruction MC_ChangeDynamic (Modifier les réglages dynamiques pour l'axe)

Tableau 10- 47 Instruction MC_ChangeDynamic

CONT/LOG	SCL	Description
	<pre>"MC_ChangeDynamic_DB"(Execute:=_bool_in_, ChangeRampUp:=_bool_in_, RampUpTime:=_real_in_, ChangeRampDown:=_bool_in_, RampDownTime:=_real_in_, ChangeEmergency:=_bool_in_, EmergencyRampTime:=_real_in_, ChangeJerkTime:=_bool_in_, JerkTime:=_real_in_, Done=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Change les paramètres dynamiques d'un axe de commande de mouvement :</p> <ul style="list-style-type: none"> • Changement de la valeur du temps d'accélération • Changement de la valeur du temps de décélération • Changement de la valeur du temps de décélération d'urgence • Changement de la valeur du temps de lissage (à-coup)

- 1 STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.
- 2 Dans l'exemple SCL, "MC_ChangeDynamic_DB" est le nom du DB d'instance.

Tableau 10- 48 Paramètres pour l'instruction MC_ChangeDynamic

Paramètre et type	Type de données	Description
Axis	IN	TO_Axis_1 Objet technologique Axe
Execute	IN	Bool Démarrage de la commande en présence d'un front montant. Valeur par défaut : FALSE
ChangeRampUp	IN	Bool TRUE = Changement du temps d'accélération conformément au paramètre d'entrée "RampUpTime". Valeur par défaut : FALSE
RampUpTime	IN	Real Temps (en secondes) pour accélérer de l'arrêt à la vitesse maximale configurée sans limite d'à-coup. Valeur par défaut : 5,00 Le changement affecte la variable <nom d'axe>. Config.DynamicDefaults.Acceleration. L'entrée en vigueur du changement est montrée dans la description de cette variable.
ChangeRampDown	IN	Bool TRUE = Changement du temps de décélération conformément au paramètre d'entrée "RampDownTime". Valeur par défaut : FALSE
RampDownTime	IN	Real Temps (en secondes) pour décélérer l'axe de la vitesse maximale configurée jusqu'à l'immobilisation sans limite d'à-coup. Valeur par défaut : 5,00 Le changement affecte la variable <nom d'axe>. Config.DynamicDefaults.Deceleration. L'entrée en vigueur du changement est montrée dans la description de cette variable.

Paramètre et type	Type de données	Description
ChangeEmergency	IN	Bool TRUE = Changement du temps de décélération d'urgence conformément au paramètre d'entrée "EmergencyRampTime" Valeur par défaut : FALSE
EmergencyRampTime	IN	Real Temps (en secondes) pour décélérer l'axe de la vitesse maximale configurée jusqu'à l'immobilisation sans limite d'à-coup en mode d'arrêt d'urgence. Valeur par défaut : 2,00 Le changement affecte la variable <nom d'axe>. Config.DynamicDefaults.EmergencyDeceleration. L'entrée en vigueur du changement est montrée dans la description de cette variable.
ChangeJerkTime	IN	Bool TRUE = Changement du temps de lissage selon le paramètre d'entrée "JerkTime". Valeur par défaut : FALSE
JerkTime	IN	Real Temps de lissage (en secondes) utilisé pour les rampes d'accélération et de décélération de l'axe. Valeur par défaut : 0,25 Le changement affecte la variable <nom d'axe>. Config.DynamicDefaults.Jerk. L'entrée en vigueur du changement est montrée dans la description de cette variable.
Done	OUT	Bool TRUE = Les valeurs modifiées ont été écrites dans le bloc de données technologique. La description des variables montre quand le changement entre en vigueur. Valeur par défaut : FALSE
Error	OUT	Bool TRUE = Une erreur s'est produite pendant l'exécution de la commande. La cause de l'erreur figure dans les paramètres "ErrorID" et "ErrorInfo". Valeur par défaut : FALSE
ErrorID	OUT	Word Code d'erreur. Valeur par défaut : 16#0000
ErrorInfo	IN	Word Informations d'erreur. Valeur par défaut : 16#0000

Conditions requises pour l'exécution de MC_ChangeDynamic :

- L'objet technologique TO_Axis_PTO V2.0 doit être correctement configuré.
- L'axe doit être débloqué.

Comportement d'annulation

Une instruction MC_ChangeDynamic ne peut être annulée par aucune autre instruction de commande de mouvement.

Une nouvelle instruction MC_ChangeDynamic n'annule aucune tâche de commande de mouvement active.

Remarque

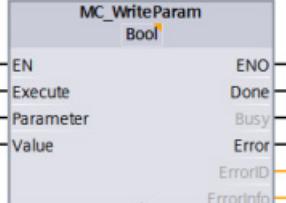
Les paramètres d'entrée "RampUpTime", "RampDownTime", "EmergencyRampTime" et "RoundingOffTime" peuvent contenir des valeurs faisant passer les paramètres d'axe résultants "accélération", "décélération", "décélération d'urgence" et "à-coup" en dehors des limites autorisées.

Veillez à maintenir les paramètres de MC_ChangeDynamic à l'intérieur des limites des paramètres de configuration dynamiques pour l'objet technologique Axe.

10.7.12 Instruction MC_WriteParam (écrire les paramètres de l'objet technologique)

Vous utilisez l'instruction MC_WriteParam pour écrire dans un nombre choisi de paramètres afin de modifier la fonctionnalité de l'axe à partir du programme utilisateur.

Tableau 10- 49 Instruction MC_WriteParam

CONT/LOG	SCL	Description
	<pre>"MC_WriteParam_DB" (Parameter:=_variant_in_, Value:=_variant_in_, Execute:=_bool_in_, Done:=_bool_out_, Error:=_real_out_, ErrorID:=_word_out_, ErrorInfo:=_word_out_);</pre>	<p>L'instruction MC_WriteParam permet d'écrire dans des paramètres publics (par exemple, accélération et valeurs du DB utilisateur).</p>

¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

² Dans l'exemple SCL, "MC_WriteParam_DB" est le nom du DB d'instance.

Vous pouvez écrire dans les paramètres publics. Vous ne pouvez pas écrire dans "MotionStatus" et "StatusBits". Les paramètres valides sont énumérés dans le tableau ci-dessous :

Nom du paramètre inscriptible	Nom du paramètre inscriptible
Actor.InverseDirection	DynamicDefaults.Acceleration
Actor.DirectionMode	DynamicDefaults.Deceleration
Actor.DriveParameter.PulsesPerDriveRevolution	DynamicDefaults.Jerk
Sensor[1].ActiveHoming.Mode	DynamicDefaults.EmergencyDeceleration
Sensor[1].ActiveHoming.SideInput	PositionLimitsHW.Active
Sensor[1].ActiveHoming.Offset	PositionLimitsHW.MaxSwitchedLevel
Sensor[1].ActiveHoming.SwitchedLevel	PositionLimitsHW.MinSwitchedLevel
Sensor[1].PassiveHoming.Mode	PositionLimitsSW.Active
Sensor[1].PassiveHoming.SideInput	PositionLimitsSW.MinPosition
Sensor[1].PassiveHoming.SwitchedLevel	PositionLimitsSW.MaxPosition
Units.LengthUnit	Homing.AutoReversal
Mechanics.LeadScrew	Homing.ApproachDirection
DynamicLimits.MinVelocity	Homing.ApproachVelocity
DynamicLimits.MaxVelocity	Homing.ReferencingVelocity

Tableau 10- 50 Paramètres pour l'instruction MC_WriteParam

Paramètre et type	Type de données	Description
PARAMNAME	IN	Variant
VALUE	IN	Variant
EXECUTE	IN	Bool
DONE	OUT	Bool
BUSY	OUT	Bool
ERROR	OUT	Real
ERRORID	OUT	Word
ERRORINFO	OUT	Word

Tableau 10- 51 Codes d'erreur pour ERRORID et ERRORINFO

ERRORID (W#16#...)	ERRORINFO (W#16#...)	Description
0	0	Modification réussie d'un paramètre du DB de l'objet technologique Axe
8410 _[1]	0028 _[1]	Réglage d'un paramètre invalide (paramètre du DB de l'objet technologique Axe de longueur incorrecte)
8410 _[1]	0029 _[1]	Réglage d'un paramètre invalide (pas de paramètre du DB de l'objet technologique Axe)
8410 _[1]	002B _[1]	Réglage d'un paramètre invalide (paramètre du DB de l'objet technologique Axe en lecture seule)
8410 _[1]	002C _[1]	Réglage d'un paramètre valide, mais l'axe n'est pas inhibé
Erreur config. _[2]	Erreur config. _[2]	Réglage d'un paramètre valide (paramètre du DB de l'objet technologique Axe public en lecture seule) hors plage
Erreur config. _[3]	Erreur config. _[3]	Réglage d'un paramètre valide (paramètre du DB de l'objet technologique Axe public) hors plage

[1] Erreur sur MC_WriteParam

[2] Erreur sur MC_Power

[3] Erreur sur MC_Power et MC_MoveXXX ou MC_CommandTable

10.7.13 Instruction MC_ReadParam (lire les paramètres de l'objet technologique)

L'instruction MC_ReadParam vous permet de lire un nombre choisi de paramètres qui indiquent la position actuelle, la vitesse, etc. de l'axe défini dans l'entrée Axe.

Tableau 10- 52 Instruction MC_ReadParam

CONT/LOG	SCL	Description
	<pre>"MC_ReadParam_DB" (Enable:=_bool_in_, Parameter:=_variant_in_, Value:=_variant_in_out_, Valid:=_bool_out_, Busy:=_bool_out_, Error:=_real_out_, ErrorID:=_word_out_, ErrorInfo:=_word_out_);</pre>	<p>L'instruction MC_ReadParam vous permet de lire des valeurs d'état individuelles, indépendamment du point de contrôle du cycle.</p>

¹ STEP 7 crée automatiquement le DB lorsque vous insérez l'instruction.

² Dans l'exemple SCL, "MC_ReadParam_DB" est le nom du DB d'instance.

L'instruction MC_ReadParam fonctionne suivant un comportement d'activation. Tant que l'entrée "Activer" est vrai, l'instruction lit le "Paramètre" indiqué dans le lieu de stockage de la "Valeur".

La valeur "MotionStatus" "Position" s'actualise à chaque point de contrôle du cycle (CCP) sur la base de la valeur HSC en cours.

La valeur "MotionStatus" "Velocity" est la vitesse de commande à la fin du segment en cours (actualisée ~10ms). MC_ReadParam peut également lire cette valeur.

Si une erreur se produit, l'instruction commute sur un état d'erreur qui ne peut être remis à zéro que par un nouveau front montant de l'entrée "Activer".

Tableau 10- 53 Paramètres pour l'instruction MC_ReadParam

Paramètre et type	Type de données	Description
ENABLE	IN	Bool Démarre l'instruction. Valeur par défaut : FALSE
PARAMETER	IN	Variant Pointeur désignant le paramètre TO à lire.
VALID	OUT	Bool Si TRUE, la valeur a été lue. Valeur par défaut : FALSE
BUSY	OUT	Bool Si TRUE, l'instruction est utilisée. Valeur par défaut : FALSE
ERROR	OUT	Real Si TRUE, une erreur s'est produite. Valeur par défaut : FALSE
ERRORID	OUT	Word Code de l'erreur. Valeur par défaut : 0
ERRORINFO	OUT	Word Information sur le ERRORID. Valeur par défaut : 0
VALUE	INOUT	Variant Pointeur désignant l'emplacement où la valeur lue est stockée.

Tableau 10- 54 Codes d'erreur pour ERRORID et ERRORINFO

ERRORID (W#16#...)	ERRORINFO (W#16#...)	Description
0	0	Un paramètre a été lu avec succès
8410	0028	Paramètre incorrect (longueur incorrecte)
8410	0029	Paramètre incorrect (pas de DB TO)
8410	0030	Paramètre incorrect (pas lisible)
8411	0032	Paramètre incorrect (valeur erronée)

Paramètres TO

La variable "MotionStatus" de l'axe peut prendre quatre valeurs. Vous souhaiterez visualiser des modifications apportées à ces valeurs lisibles pendant l'exécution du programme :

Nom de variable	Type de données	Lisible via MC_ReadParam
MotionStatus :	Structure	Non
• Position	REAL	Oui
• Vitesse	REAL	Oui
• Distance	REAL	Oui
• TargetPosition	REAL	Oui

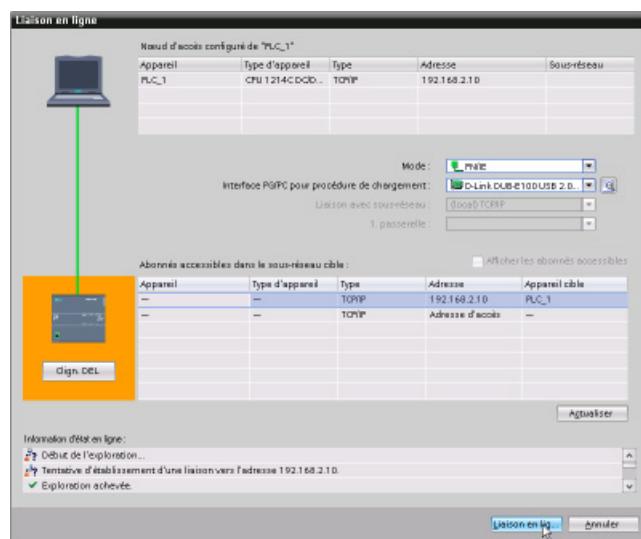
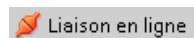
Utilisation aisée des outils en ligne

11.1 Passage en ligne et connexion à une CPU

Vous devez établir une liaison en ligne entre la console de programmation et la CPU pour charger des programmes et des données techniques de projet, ainsi que pour des activités telles que les suivantes :

- Test des programmes utilisateur
- Affichage et changement de l'état de fonctionnement de la CPU (Page 354)
- Affichage et réglage de la date et de l'heure de la CPU (Page 365)
- Affichage des informations sur les modules
- Comparaison et synchronisation (Page 364) hors ligne par rapport à des blocs de programme en ligne
- Chargement de blocs de programme depuis la CPU et dans la CPU
- Affichage du diagnostic et de la mémoire tampon de diagnostic (Page 365)
- Utilisation d'une table de visualisation (Page 357) pour tester le programme utilisateur par visualisation et forçage de valeurs
- Utilisation d'une table de forçage permanent pour forcer des valeurs de manière permanente dans la CPU (Page 359)

Pour établir une liaison en ligne à une CPU configurée, cliquez sur la CPU dans l'arborescence du projet et cliquez sur le bouton "Liaison en ligne" dans la vue du projet :



Si c'est la première fois que vous allez en ligne avec cette CPU, vous devez sélectionner le type d'interface PG/PC et l'interface PG/PC spécifique dans la boîte de dialogue Liaison en ligne avant d'établir une liaison en ligne à une CPU trouvée sur cette interface.

Vous avez maintenant connecté votre console de programmation à la CPU. Les cadres de couleur orange correspondent à une liaison en ligne. Vous pouvez désormais utiliser les outils en ligne et de diagnostic dans l'arborescence du projet et dans la Task Card Outils en ligne.

11.2 Interaction avec la CPU en ligne

La Task Card "Outils en ligne" affiche dans la vue du projet un pupitre opérateur indiquant l'état de fonctionnement de la CPU en ligne. Ce pupitre opérateur vous permet également de changer l'état de fonctionnement de la CPU en ligne. Servez-vous du bouton sur le pupitre opérateur pour changer l'état de fonctionnement (ARRET ou MARCHE). Le pupitre opérateur dispose également d'un bouton MRES pour réinitialiser la mémoire.

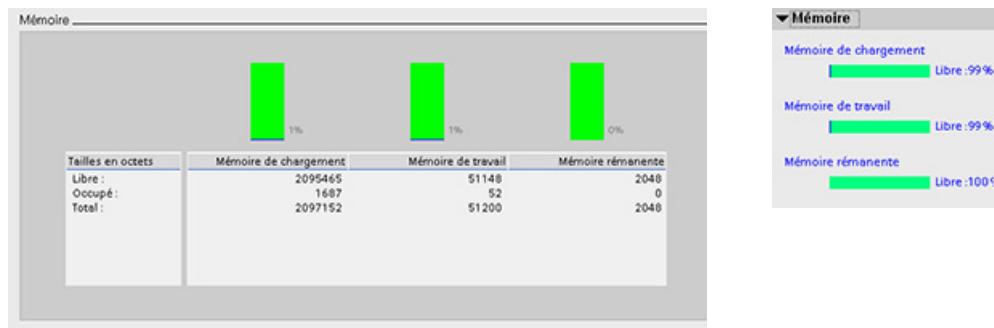


La couleur de l'indicateur MARCHE/ARRET signale l'état de fonctionnement en cours de la CPU : le jaune indique l'état ARRET et le vert l'état MARCHE.

Pour utiliser le pupitre opérateur, vous devez établir une connexion en ligne entre STEP 7 et la CPU. Après avoir sélectionné la CPU dans la configuration de l'appareil ou affiché un bloc de code dans la CPU en ligne, vous pouvez afficher le pupitre opérateur à partir de la Task Card "Outils en ligne".



Vous pouvez surveiller le temps de cycle d'une CPU en ligne.

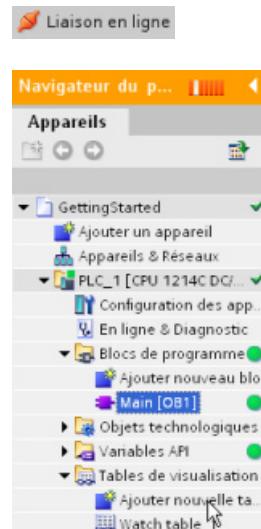


Vous pouvez également visualiser l'utilisation de la mémoire de la CPU.

11.3

Passage en ligne pour visualiser les valeurs dans la CPU

Pour visualiser les variables, une liaison en ligne doit être établie à la CPU. Il vous suffit de cliquer sur le bouton "Liaison en ligne" de la barre d'outils.



Une fois que vous êtes connecté à la CPU, STEP 7 affiche les en-têtes des zones de travail en orange.

L'arborescence du projet affiche une comparaison du projet hors ligne et de la CPU en ligne. Un cercle vert signifie que la CPU et le projet sont synchronisés, c'est-à-dire que tous deux ont la même configuration et le même programme utilisateur.

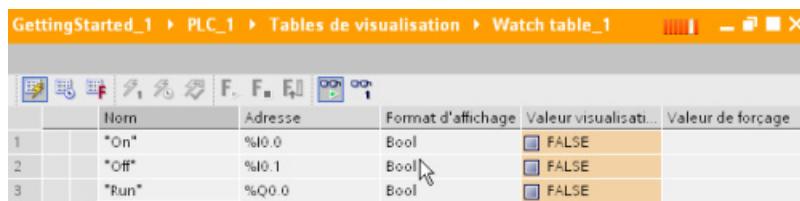
Les tables de variables montrent les variables. Les tables de visualisation peuvent également montrer les variables, ainsi que les adresses directes.

	Nom	Adresse	Format d'affichage	Valeur visualisati...	Valeur de forçage
1	*On*	%I0.0	Bool		
2	*Off*	%I0.1	Bool		
3	*Run*	%Q0.0	Bool		

11.4 L'affichage de l'état du programme utilisateur est simple



Pour visualiser l'exécution du programme utilisateur et afficher les valeurs des variables, cliquez sur le bouton "Visualiser tout" de la barre d'outils.



	Nom	Adresse	Format d'affichage	Valeur visualisati...	Valeur de forçage
1	"On"	%I0.0	Bool	FALSE	
2	"Off"	%I0.1	Bool	FALSE	
3	"Run"	%Q0.0	Bool	FALSE	

Le champ "Valeur de visualisation" indique la valeur de chaque variable.

11.4 L'affichage de l'état du programme utilisateur est simple

Vous pouvez visualiser l'état d'un maximum de 50 variables dans les éditeurs de programme CONT et LOG. Utilisez la barre des éditeurs pour afficher l'éditeur CONT. La barre des éditeurs vous permet de basculer entre les éditeurs ouverts sans avoir à les ouvrir ou les fermer.

Dans la barre d'outils de l'éditeur de programme, cliquez sur le bouton "Marche/arrêt visualisation" pour afficher le statut de votre programme utilisateur.



Le réseau dans l'éditeur de programme affiche le flux de courant en vert.

Vous pouvez également faire un clic droit sur l'opération ou le paramètre pour modifier la valeur pour l'opération.

11.5 Utilisation d'une table de visualisation pour surveiller la CPU

Une table de visualisation vous permet de visualiser et de forcer des données pendant que la CPU exécute votre programme utilisateur. Il peut s'agir d'entrées (I), de sorties (Q), de mémentos M, d'un DB ou d'entrées ou de sorties de périphérie (telles que "On:P", "I 3.4:P" ou "Q3.4:P"). Vous ne pouvez pas visualiser les sorties physiques (comme Q0.0:P) avec précision, car la fonction de visualisation peut uniquement afficher la dernière valeur écrite depuis la mémoire Q et ne lit pas la valeur réelle dans les sorties physiques.

La fonction de visualisation ne modifie pas la séquence du programme. Elle vous donne des informations sur la séquence du programme et les données du programme dans la CPU. Vous pouvez également utiliser la fonction "Forçage" pour tester l'exécution de votre programme utilisateur.



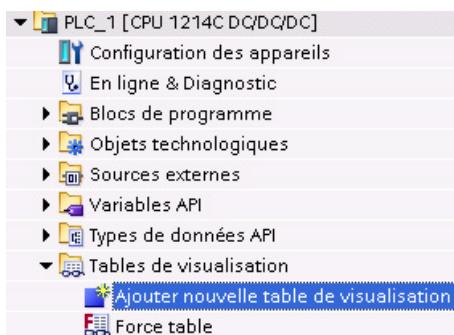
	Nom	Adresse	Format d'affich...	Valeur visualisation	Visualisation avec décl...	Forçage avec décl...	Valeur de forçage
1	"Start"	%I0.0	Bool		Permanent	Permanent	<input type="checkbox"/>
2	"Stop"	%I0.1	Bool		Permanent	Permanent	<input type="checkbox"/>
3	"Running"	%M0.0	Bool		Permanent	Permanent	<input type="checkbox"/>

Remarque

Les E/S TOR utilisées par le compteur rapide (HSC), la modulation de largeur d'impulsion (PWM) et la sortie de trains d'impulsions (PTO) sont affectées pendant la configuration de ces appareils. Lorsque des adresses d'E/S TOR sont affectées à ces appareils, les valeurs des adresses d'E/S affectées ne peuvent pas être forcées au moyen de la fonction de "Forçage permanent" de la table de visualisation.

Une table de visualisation permet de visualiser ou de forcer les valeurs de variables individuelles, après avoir choisi parmi les options suivantes :

- Au début ou à la fin du cycle
- Quand la CPU passe à l'état ARRET
- "En permanence" (la valeur n'est pas remise à 0 après une transition de ARRET à MARCHE.)



Pour créer une table de visualisation :

1. Double-cliquez sur "Ajouter nouvelle table de visualisation" pour ouvrir une nouvelle table de visualisation.
2. Entrez un nom de variable pour ajouter une variable à la table de visualisation.

Pour visualiser les variables, une liaison en ligne doit être établie à la CPU. Les options suivantes sont disponibles pour le forçage des variables :

- "Forçage immédiat" modifie immédiatement la valeur des adresses sélectionnées pour un cycle.
- "Forcer avec déclenchement" modifie les valeurs des adresses sélectionnées.

Cette fonction ne fournit pas de retour pour indiquer que les adresses sélectionnées ont été effectivement forcées. Si vous avez besoin d'un retour, utilisez la fonction "Forçage immédiat".

- "Débloquer sorties de périphérie" vous permet d'activer les sorties de périphérie lorsque la CPU est à l'état ARRET. Cette caractéristique est utile pour le test du câblage des modules de sortie.

Vous pouvez sélectionner les diverses fonctions à l'aide des boutons en haut de la table de visualisation. Entrez le nom de la variable à visualiser et sélectionnez un format d'affichage dans la liste de sélection déroulante. Lorsqu'une liaison en ligne est établie avec la CPU, un clic sur le bouton "Visualiser" affiche la valeur en cours de la donnée dans le champ "Valeur de visualisation".

11.6 Utilisation de la table de forçage permanent

Une table de forçage permanent fournit une fonction de "forçage permanent" qui écrase la valeur d'une entrée ou d'une sortie avec une valeur spécifiée pour l'adresse de périphérie d'entrée ou de sortie. La CPU applique cette valeur forcée en permanence à la mémoire image des entrées avant l'exécution du programme utilisateur et à la mémoire image des sorties avant l'écriture des sorties dans les modules.

Remarque

Les valeurs de forçage permanent sont stockées dans la CPU et non dans la table de forçage permanent.

Vous ne pouvez pas forcer une entrée (adresse I) ou une sortie (adresse Q), mais vous pouvez forcer une entrée ou une sortie de périphérie. La table de forçage permanent ajoute automatiquement ":P" à l'adresse (par exemple, "On":P ou "Run":P).

	i	Nom	Adresse	Format d'affichage	Valeur visualisation	Valeur de forçage permanent	F
1		"On":P	%I0.0:P	Bool		TRUE	<input checked="" type="checkbox"/> 
2		"Off":P	%I0.1:P	Bool			<input type="checkbox"/>
3		"Run":P	%Q0.1:P	Bool			<input type="checkbox"/>

Entrez la valeur pour l'entrée ou la sortie à forcer dans la cellule "Valeur de forçage permanent". Vous pouvez alors utiliser la case à cocher dans la colonne "Forçage permanent" pour activer le forçage permanent de l'entrée ou de la sortie.

 Utilisez le bouton "Démarrer ou remplacer forçage permanent" pour forcer la valeur des variables de manière permanente dans la table de forçage permanent. Cliquez sur le bouton "Arrêter forçage permanent" pour arrêter le forçage des variables.

Dans la table de forçage permanent, vous pouvez visualiser l'état de la valeur forcée pour une entrée, mais vous ne pouvez pas visualiser la valeur forcée d'une sortie.

Vous pouvez également visualiser l'état des variables forcées dans l'éditeur de programmes.



Remarque

Lors du forçage permanent d'une entrée ou d'une sortie dans une table de forçage permanent, les actions de forçage permanent deviennent partie intégrante de la configuration du projet. Si vous fermez STEP 7, les éléments forcés de manière permanente restent actifs dans le programme de la CPU, et ce jusqu'à ce qu'ils soient effacés. Pour annuler le forçage permanent de ces éléments, vous devez utiliser STEP 7 pour vous connecter à la CPU en ligne, puis utiliser la table de forçage permanent pour désactiver ou arrêter la fonction de forçage permanent pour ces éléments.

Vous pouvez effectuer un forçage permanent d'entrées et de sorties en indiquant l'adresse d'entrée ou de sortie physique (I_:P ou Q_:P) dans la table de forçage permanent puis en lançant la fonction de forçage permanent.

Dans le programme, les entrées physiques lues sont remplacées par les valeurs de forçage permanent. Le programme utilise les valeurs de forçage permanent pendant le traitement. Lorsque le programme écrit dans une sortie physique, la valeur de sortie est remplacée par la valeur de forçage permanent. La valeur de forçage permanent apparaît au niveau de la sortie physique et est utilisée par le processus.

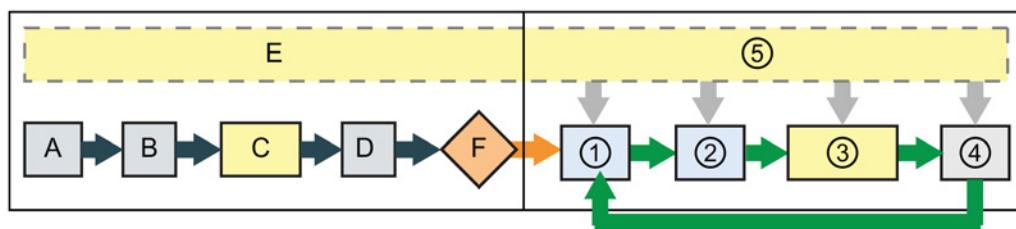
Lors du forçage permanent d'une entrée ou d'une sortie dans la table de forçage permanent, les actions de forçage deviennent partie intégrante du programme utilisateur. Même si vous fermez le logiciel de programmation, les sélections de forçage permanent restent actives dans le programme CPU qui fonctionne, jusqu'à ce que vous les annuliez en commutant en ligne avec le logiciel de programmation et en arrêtant la fonction de forçage permanent. Les programmes à entrées ou sorties forcées qui sont chargés dans une autre CPU à partir d'une carte mémoire continuent à appliquer le forçage permanent aux entrées et sorties sélectionnées dans le programme.

Si la CPU exécute le programme utilisateur à partir d'une carte mémoire protégée en écriture, vous ne pouvez pas lancer ou modifier le forçage permanent d'E/S à partir d'une table de visualisation, car il n'est pas possible d'écraser les valeurs dans le programme utilisateur protégé en écriture. Toute tentative de forçage permanent de valeurs protégées en écriture génère une erreur. Si vous utilisez une carte mémoire pour transférer un programme utilisateur, tout élément forcé en permanence sur cette carte mémoire sera transféré dans la CPU.

Remarque

Les E/S TOR affectées à HSC, PWM et PTO ne peuvent pas être forcées.

Les E/S TOR utilisées par le compteur rapide (HSC), la modulation de largeur d'impulsion (PWM) et la sortie de trains d'impulsions (PTO) sont affectées pendant la configuration des appareils. Lorsque des adresses d'E/S TOR sont affectées à ces appareils, les valeurs dans les adresses d'E/S affectées ne peuvent pas être forcées par la fonction de forçage permanent de la table de forçage permanent.



Mise en route

- A L'effacement de la zone de mémoire I n'est pas affecté par la fonction Forçage permanent.
- B L'initialisation des valeurs de sortie n'est pas affectée par la fonction Forçage permanent.
- C Pendant l'exécution des OB de démarrage, la CPU applique la valeur de forçage permanent lorsque le programme utilisateur accède à l'entrée physique.
- D L'enregistrement d'événements d'alarme dans la file d'attente n'est pas affecté.
- E La validation de l'écriture dans les sorties n'est pas affectée.

MARCHE

- ① Pendant l'écriture de la mémoire Q dans les sorties physiques, la CPU applique la valeur de forçage permanent lors de la mise à jour des sorties.
- ② Lors de la lecture des entrées physiques, la CPU applique les valeurs de forçage permanent juste avant de copier les entrées en mémoire I.
- ③ Pendant l'exécution du programme utilisateur (OB de cycle de programme), la CPU applique la valeur de forçage permanent lorsque le programme utilisateur accède à l'entrée physique ou écrit dans la sortie physique.
- ④ Le traitement des demandes de communications et le test d'auto-diagnostic ne sont pas affectés par la fonction Forçage permanent.
- ⑤ Le traitement des alarmes à n'importe quel moment du cycle n'est pas affecté.

11.7 Acquisition des valeurs en ligne d'un DB pour redéfinir les valeurs initiales

Vous pouvez capturer les valeurs en cours visualisées dans une CPU en ligne pour en faire les valeurs initiales d'un DB global.

- Une liaison en ligne doit être établie à la CPU.
- La CPU doit être à l'état MARCHE.
- Le DB doit être ouvert dans STEP 7.



Utilisez le bouton "Afficher un instantané des valeurs de visualisation" pour capturer les valeurs en cours des variables sélectionnées dans le DB. Vous pouvez alors copier ces valeurs dans la colonne "Valeur initiale" du DB.

1. Dans l'éditeur de DB, cliquez sur le bouton "Visualiser tout". La colonne "Valeur de visualisation" affiche les valeurs de données en cours.
2. Cliquez sur le bouton "Afficher un instantané des valeurs de visualisation" pour afficher les valeurs en cours dans la colonne "Instantané".
3. Cliquez sur le bouton "Visualiser tout" pour arrêter la visualisation des données dans la CPU.
4. Copiez une valeur de la colonne "Instantané" pour une variable.
 - Sélectionnez la valeur à copier.
 - Cliquez avec le bouton droit de la souris sur la valeur sélectionnée afin d'afficher le menu contextuel.
 - Sélectionnez la commande "Copier".
5. Collez la valeur copiée dans la colonne "Valeur initiale" correspondante de la variable (cliquez avec le bouton droit de la souris sur la cellule et sélectionnez "Coller" dans le menu contextuel).
6. Enregistrez le projet pour configurer les valeurs copiées en tant que nouvelles valeurs initiales pour le DB.
7. Compilez et chargez le DB dans la CPU Le DB utilisera les nouvelles valeurs initiales après le passage de la CPU à l'état MARCHE.

Remarque

Les valeurs visibles dans la colonne "Valeur de visualisation" sont toujours copiées de la CPU. STEP 7 ne vérifie pas si toutes les valeurs proviennent du même cycle de la CPU.

11.8 Chargement des éléments du projet

Vous pouvez également copier les blocs de programme depuis une CPU en ligne ou une carte mémoire reliée à votre console de programmation.

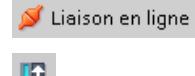
Préparez le projet hors ligne pour les blocs de programme copiés :

1. Ajoutez un appareil CPU qui correspond à la CPU en ligne.
2. Cliquez une fois sur le noeud de la CPU pour rendre le dossier "Blocs de programme" visible.

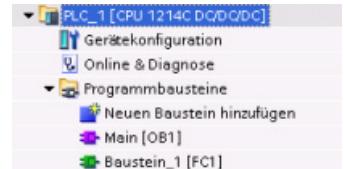


Procédez comme suit pour charger les blocs de programme de la CPU en ligne dans le projet hors ligne :

1. Cliquez sur le dossier "Blocs de programme" dans le projet hors ligne.
2. Cliquez sur le bouton "Liaison en ligne".
3. Cliquez sur le bouton "Charger depuis la CPU".
4. Confirmez votre décision dans la boîte de dialogue Charger depuis la CPU (Page 353).

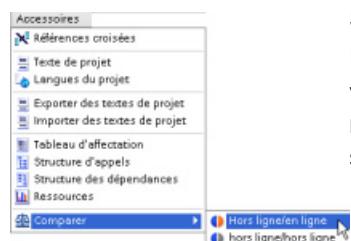


Une fois le chargement terminé, STEP 7 affiche tous les blocs de programmes chargés dans le projet.



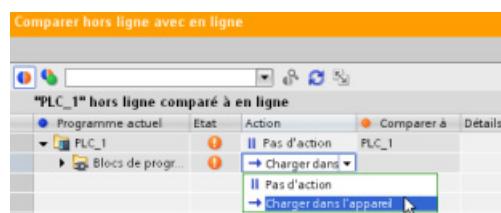
11.9 Comparaison de CPU hors ligne et en ligne

Vous pouvez comparer les blocs de code dans une CPU en ligne avec les blocs de code de votre projet. Si les blocs de code de votre projet ne correspondent pas à ceux de la CPU en ligne, l'éditeur de comparaison vous permet de synchroniser votre projet avec la CPU en ligne soit en chargeant les blocs de code de votre projet dans la CPU, soit en effaçant les blocs du projet qui n'existent pas dans la CPU en ligne.



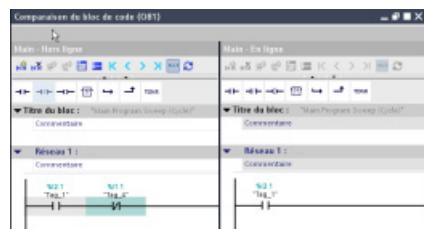
Sélectionnez la CPU dans votre projet.

Utilisez la commande "Comparer hors ligne/en ligne" pour ouvrir l'éditeur "Comparer" (accédez à la commande soit depuis le menu "Outils", soit en cliquant avec le bouton droit de la souris sur la CPU dans votre projet).



Cliquez dans la colonne "Action" d'un objet et choisissez de supprimer l'objet, de ne rien faire ou de charger l'objet dans l'appareil.

Cliquez sur le bouton "Synchroniser" pour charger les blocs de code.



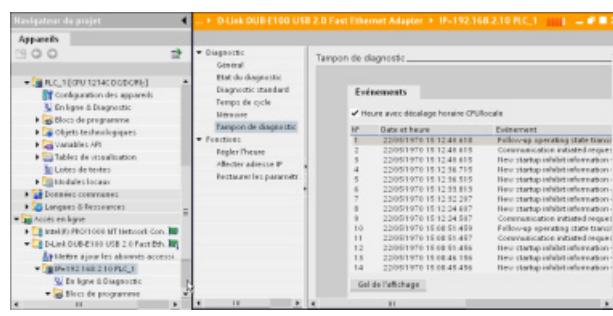
Cliquez avec le bouton droit de la souris sur un objet dans la colonne "Comparer à" et cliquez sur le bouton "Démarrer la comparaison détaillée" pour afficher les blocs de code côté à côté.

La comparaison détaillée met en évidence les différences entre les blocs de code de la CPU en ligne et les blocs de code de la CPU dans votre projet.

11.10 Affichage des événements de diagnostic

La CPU met à disposition un tampon de diagnostic qui contient une entrée pour chaque événement de diagnostic, tel qu'une commutation de l'état de fonctionnement de la CPU ou les erreurs détectées dans la CPU ou dans les modules. Vous devez être en ligne pour accéder à la mémoire tampon de diagnostic.

Chaque entrée inclut la date et l'heure auxquelles l'événement s'est produit, une catégorie d'événement, ainsi qu'une description de l'événement. Les entrées sont affichées dans l'ordre chronologique avec l'événement le plus récent en haut.



Tant que la CPU est sous tension, jusqu'à 50 événements les plus récents sont disponibles dans ce journal. Lorsque le journal est plein, l'événement le plus ancien dans le journal est remplacé par un nouvel événement.

En cas de coupure de courant, les dix événements les plus récents sont sauvegardés.

11.11 Réglage de l'adresse IP et de l'heure

Vous pouvez régler l'adresse IP et l'heure dans la CPU en ligne. Après avoir accédé à "En ligne et Diagnostics" depuis l'arborescence du Projet pour une CPU en ligne, vous pouvez afficher ou modifier l'adresse IP. Vous pouvez également afficher ou régler les paramètres de date et d'heure de la CPU en ligne.



Remarque

Cette fonction est disponible uniquement pour une CPU qui soit dispose uniquement d'une adresse MAC (pas encore d'adresse IP attribuée), soit a été réinitialisée aux réglages d'usine.

11.12 Restauration des réglages d'usine

Vous pouvez restaurer les réglages d'usine d'origine d'un S7-1200 dans les conditions suivantes :

- La CPU dispose d'une liaison en ligne.
- La CPU est à l'état ARRET.

Remarque

Si la CPU est à l'état MARCHE et que vous lancez l'opération de restauration, vous pourrez la faire passer à l'état ARRET après acquittement d'un message de confirmation.

Marche à suivre

Procédez comme suit pour restaurer les réglages d'usine d'une CPU :

1. Ouvrez la vue En ligne & Diagnostic de la CPU.
2. Sélectionnez "Restaurer les réglages d'usine" dans le dossier "Fonctions".
3. Cochez la case "Conserver l'adresse IP" si vous voulez conserver l'adresse IP ou "Effacer l'adresse IP" si vous voulez effacer l'adresse IP.
4. Cliquez sur le bouton "Restaurer".
5. Acquittez le message de confirmation en cliquant sur "OK".

Résultat

Le module passe à l'état ARRET si nécessaire et les réglages d'usine sont restaurés. La CPU effectue les actions suivantes :

si une carte mémoire est installée dans la CPU	si aucune carte mémoire n'est installée dans la CPU
<ul style="list-style-type: none">Efface la mémoire tampon de diagnosticRéinitialise l'heureRestaure la mémoire de travail de la carte mémoireMet toutes les zones d'opérandes aux valeurs initiales configuréesMet tous les paramètres aux valeurs configuréesConserve ou efface l'adresse IP en fonction de la sélection effectuée. (L'adresse MAC est fixe et n'est jamais modifiée)¹.Supprime l'enregistrement des données de commande, le cas échéant	<ul style="list-style-type: none">Efface la mémoire tampon de diagnosticRéinitialise l'heureEfface la mémoire de travail et la mémoire de chargement interneMet toutes les zones d'opérandes aux valeurs initiales configuréesMet tous les paramètres aux valeurs configuréesConserve ou efface l'adresse IP en fonction de la sélection effectuée. (L'adresse MAC est fixe et n'est jamais modifiée)¹.Supprimer l'enregistrement des données de commande, le cas échéant

¹ Si vous avez sélectionné "Conserver l'adresse IP", la CPU règle l'adresse IP, le masque de sous-réseau et l'adresse du routeur (le cas échéant) selon les paramètres de votre configuration matérielle, à moins que vous n'ayez modifié ces valeurs dans le programme utilisateur ou un autre outil, auquel cas la CPU restaure les valeurs modifiées.

11.13 Mise à jour du firmware

Vous pouvez actualiser le firmware de la CPU connectée dans les outils en ligne et de diagnostic de STEP 7.

Procédez comme suit pour mettre à jour le firmware :

1. Ouvrez la vue En ligne & Diagnostic de la CPU connectée.
2. Sélectionnez "Mise à jour du firmware" dans le dossier "Fonctions".
3. Cliquez sur le bouton Naviguer et naviguez jusqu'à l'emplacement qui contient le fichier de mise à jour du logiciel. Il peut s'agir d'un emplacement sur votre disque dur où vous avez téléchargé un fichier de mise à jour du firmware S7-1200 (<http://support.automation.siemens.com/WW/view/fr/34612486/133100>) à partir du site Web de service et de support (<http://www.siemens.com/automation/>).
4. Sélectionnez un fichier qui est compatible avec votre module. Le tableau présente les modules compatibles correspondant au fichier sélectionné.
5. Cliquez sur le bouton "Démarrer mise à jour". Suivez les boîtes de dialogue, si nécessaire, pour changer l'état de fonctionnement de votre CPU.

STEP 7 affiche des boîtes de dialogue d'avancement pendant qu'il charge la mise à jour du logiciel. Lorsqu'il a terminé, il vous demande de démarrer le module avec le nouveau firmware.

Remarque

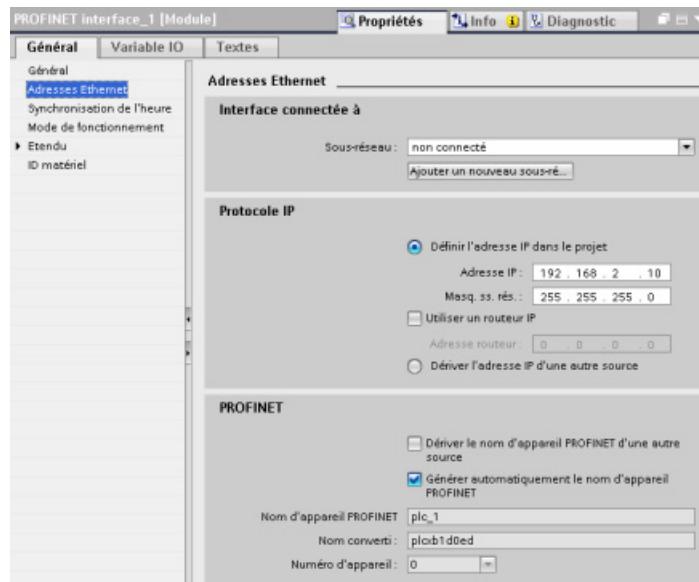
Si vous décidez de ne pas démarrer le module avec le nouveau firmware, le firmware précédent reste actif jusqu'à ce que vous réinitialisiez le module, par exemple par mise hors tension puis sous tension. Le nouveau firmware ne devient actif qu'après réinitialisation du module.

Vous pouvez également mettre à jour le firmware au moyen de l'une des autres méthodes suivantes :

- Utilisation d'une carte mémoire (Page 62)
- Utilisation de la page Web standard "Information sur les modules" du serveur Web (Page 264)

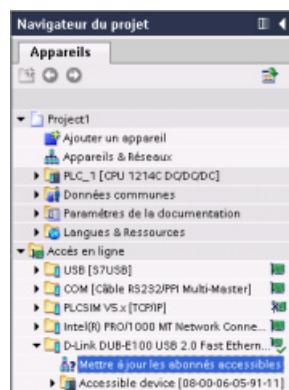
11.14 Chargement d'une adresse IP permanente dans une CPU en ligne

Pour affecter une adresse IP, vous devez exécuter les tâches suivantes :



- Configurez l'adresse IP de la CPU (Page 87).
- Enregistrez et chargez la configuration dans la CPU.

L'adresse IP et le masque de sous-réseau de la CPU doivent être compatibles avec l'adresse IP et le masque de sous-réseau de la console de programmation. Adressez-vous à votre administrateur réseau pour obtenir l'adresse IP et le masque de sous-réseau de votre CPU.



Si la CPU n'a pas été configurée auparavant, vous pouvez également utiliser l'"Accès en ligne" pour régler l'adresse IP.

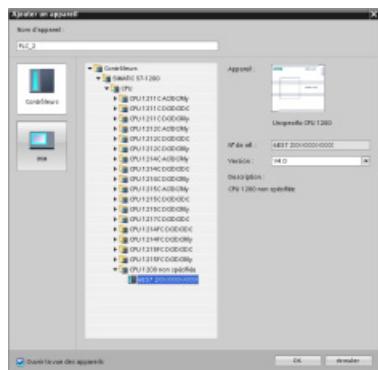
Une adresse IP que vous avez chargée avec la configuration de l'appareil n'est pas perdue en cas de mise hors tension puis mise sous tension de l'API.

Après avoir chargé la configuration de l'appareil avec l'adresse IP, vous pouvez voir l'adresse IP dans le dossier "Accès en ligne".

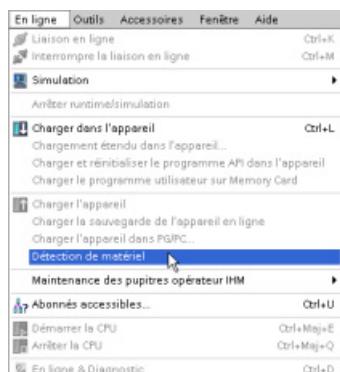
11.15 Utilisation de la "CPU non spécifiée" pour charger la configuration matérielle

Si vous disposez d'une CPU physique que vous pouvez connecter à la console de programmation, il est facile d'y charger la configuration matérielle.

Vous devez d'abord connecter la CPU à la console de programmation, puis créer un nouveau projet.

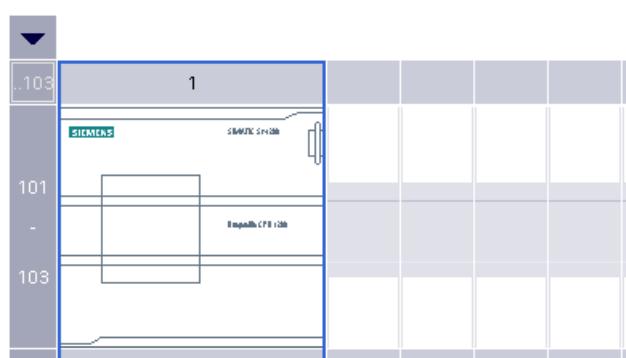


Dans la configuration de l'appareil (vue Projet ou vue Portail), ajoutez un nouvel appareil, mais sélectionnez "CPU non spécifiée" au lieu de sélectionner une CPU spécifique. STEP 7 crée une CPU non spécifiée.



Après avoir créé la CPU non spécifiée, vous pouvez charger la configuration matérielle dans la CPU en ligne.

- Dans l'éditeur de programmes, vous sélectionnez la commande "Détection du matériel" du menu "En ligne".
- Dans l'éditeur de configuration des appareils, vous sélectionnez l'option permettant de détecter la configuration de l'appareil connecté.



Après avoir sélectionné la CPU à partir de la boîte de dialogue en ligne, STEP 7 télécharge la configuration matérielle depuis la CPU, en incluant tous les modules (SM, SB ou CM). L'adresse IP n'est **pas** chargée. Vous devez configurer l'adresse IP manuellement dans "Configuration de l'appareil".

11.16 Chargement dans la CPU à l'état MARCHE

La CPU prend en charge la fonction "Chargement dans la CPU à l'état MARCHE". Cette fonction doit vous permettre d'apporter de petites modifications au programme utilisateur en entraînant le moins de gêne possible pour le processus commandé par le programme. La réalisation de cette fonction permet toutefois des modifications importantes du programme, pouvant provoquer des perturbations ou même s'avérer dangereuses.

ATTENTION

Risques liés au chargement dans la CPU à l'état MARCHE

Lorsque vous chargez des modifications dans la CPU à l'état MARCHE, ces modifications affectent immédiatement le fonctionnement du processus. La modification du programme à l'état MARCHE peut entraîner un fonctionnement inattendu du système, pouvant causer la mort, des blessures graves ou des dégâts matériels importants.

Aussi, seules des personnes autorisées, conscientes des conséquences des modifications à l'état MARCHE sur le fonctionnement du système, sont-elles habilitées à effectuer un chargement à l'état MARCHE.

La fonction "Chargement à l'état MARCHE" permet de modifier un programme et de charger les modifications dans votre CPU sans faire passer cette dernière à l'état ARRET :

- Vous pouvez apporter des modifications mineures au processus en cours sans avoir à l'arrêter (par exemple, modifier la valeur d'un paramètre).
- Vous pouvez déboguer un programme plus rapidement grâce à cette fonction (par exemple, inverser la logique d'un contact normalement ouvert ou fermé).

Vous pouvez apporter les modifications suivantes aux variables et blocs de programme et les charger dans la CPU à l'état MARCHE :

- Créer, remplacer et effacer des fonctions (FC), blocs fonctionnels (FB) et tables de variables.
- Créer, effacer et remplacer des blocs de données (DB) et des blocs de données d'instance pour blocs fonctionnels (FB). Vous pouvez procéder à des ajouts dans des structures de DB et les charger dans la CPU à l'état MARCHE. Selon vos paramétrages de configuration, la CPU peut conserver les valeurs des variables de DB existantes et initialiser les nouvelles variables du DB à leur valeur initiale ou bien elle peut réinitialiser toutes les variables du DB à leur valeur initiale. Vous ne pouvez pas charger un DB du serveur Web (commande ou fragment) à l'état MARCHE.
- Remplacer des blocs d'organisation (OB). Vous ne pouvez toutefois ni créer d'OB, ni en supprimer.

Vous pouvez charger simultanément jusqu'à vingt blocs à l'état MARCHE. Si vous devez charger plus de vingt blocs, vous devez faire passer la CPU à l'état ARRET.

Si vous chargez des modifications dans un processus réel (et non dans un processus simulé comme cela pourrait être le cas au cours du débogage d'un programme), il est vital de considérer toutes les conséquences en termes de sécurité sur les machines et les opérateurs des machines avant le chargement.

Remarque

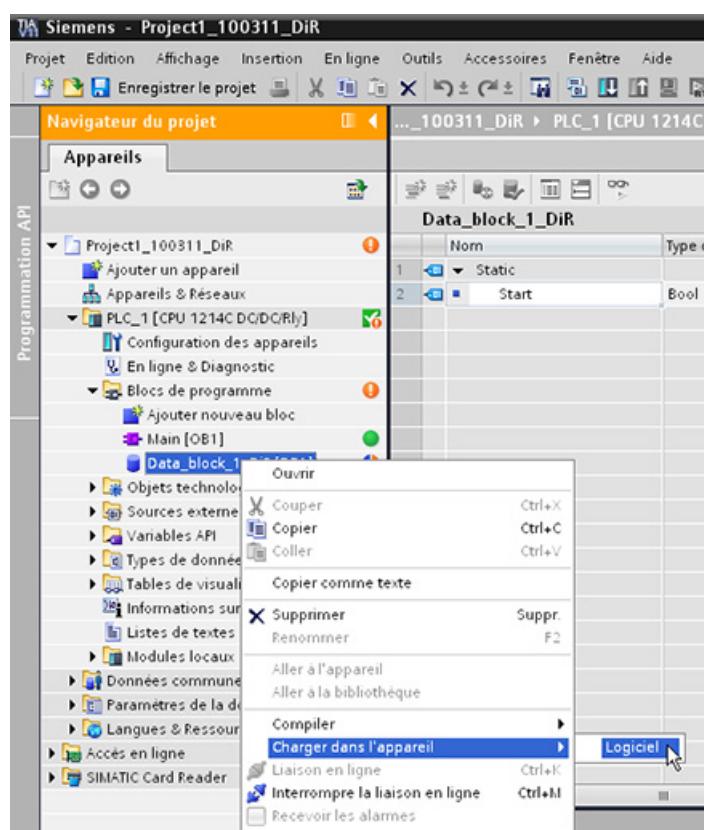
Si la CPU est à l'état MARCHE et que des modifications ont été apportées au programme, STEP 7 tentera toujours, dans un premier temps, d'effectuer un chargement dans la CPU à l'état MARCHE. Si ce n'est pas ce que vous désirez, vous devez faire passer la CPU à l'état ARRET.

Si les modifications effectuées ne sont pas acceptées en chargement à l'état MARCHE, STEP 7 demandera à l'utilisateur de mettre la CPU à l'état ARRET.

11.16.1 Modification du programme à l'état "Marche"

Pour modifier le programme à l'état MARCHE, vous devez d'abord vous assurer que la CPU et le programme remplissent les conditions requises, puis exécuter la procédure suivante :

1. Sélectionnez l'une des méthodes suivantes pour charger votre programme dans la CPU à l'état MARCHE :
 - Sélectionnez la commande "Charger dans l'appareil" du menu "En ligne".
 - Cliquez sur le bouton "Charger dans l'appareil" de la barre d'outils.
 - Clic droit sur "Blocs de programme" dans le navigateur du projet et sélection de la commande "Charger dans l'appareil > Logiciel"



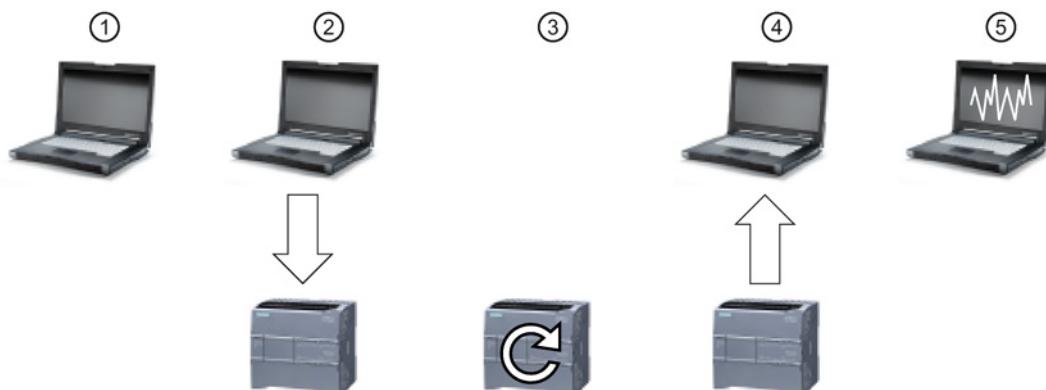
Si la compilation du programme s'exécute sans erreur, STEP 7 commence le chargement du programme dans la CPU.

2. Lorsque STEP 7 vous demande de charger votre programme ou d'annuler l'opération, cliquez sur "Charger" pour charger le programme dans la CPU.

11.17 Traçage et enregistrement de données CPU en fonctions de conditions de déclenchement

STEP 7 fournit des fonctions de traçage et d'analyse logique qui vous permettent de configurer des variables que l'API doit tracer et enregistrer. Vous pouvez alors charger les données de traçage enregistrées dans votre console de programmation et utiliser les outils STEP 7 pour analyser, gérer et représenter graphiquement vos données. Vous utilisez le dossier Traces dans l'arborescence de projet STEP 7 pour créer et gérer les traces.

La figure suivante montre les différentes étapes de la fonction de traçage :



- ① Configurez la trace dans l'éditeur de trace de STEP 7. Vous pouvez configurer les valeurs de données à enregistrer, la durée et la fréquence d'enregistrement, ainsi que la condition de déclenchement.
- ② Transférez la configuration de la trace de STEP 7 dans l'API.
- ③ L'API exécute le programme et, lorsque la condition de déclenchement se produit, il déclenche l'enregistrement des données de trace.
- ④ Transférez les valeurs enregistrées de l'API dans STEP 7.
- ⑤ Utilisez les outils de STEP 7 pour analyser les données, les afficher graphiquement et les enregistrer.

La taille maximum d'un traçage est de 512 Koctets par traçage.

Accès aux exemples

Consultez le système d'information de STEP 7 pour plus de détails sur la manière de programmer une trace, de charger la configuration dans la CPU, de charger les données de traçage depuis la CPU et d'afficher les données dans l'analyseur logique. Vous trouverez ici des exemples détaillées dans le chapitre "Utilisation de fonctions en ligne et de diagnostic > Utilisation de fonctions de traçage et d'analyse logique".

De plus, le manuel en ligne "Industry Automation SINAMICS/SIMATIC Utilisation de fonctions de traçage et d'analyse logique" (<http://support.automation.siemens.com/WW/view/fr/64897128>) est une bonne référence.

12.1

Vue d'ensemble de la technologie IO-Link

IO-Link est une technologie de communication innovante pour des capteurs et des actionneurs définis par l'organisation de l'utilisateur de PROFIBUS (PNO). IO-Link est une norme internationale conforme à IEC 61131-9. Elle repose sur une connexion point-à-point entre les capteurs et les actionneurs (esclaves) et le contrôleur (maître). Elle ne représente donc pas un système de bus mais est une mise à jour de la connexion conventionnelle point-à-point.

En plus des données de fonctionnement cyclique, des données étendues de paramètres et de diagnostic sont transmises par les capteurs/actionneurs connectés. On utilise pour la transmission de données le même câble de connexion à 3 fils qui est utilisé pour la technologie de capteur standard.

12.2

Composants d'un système IO-Link

Un système IO-Link se compose d'appareils IO-Link (habituellement des capteurs, des actionneurs ou une combinaison des deux), d'un câble de capteur/actionneur standard à 3 fils et d'un maître IO-Link. Le maître peut être un appareil avec n'importe quelle conception et n'importe quel niveau de protection.

Un maître IO-Link peut avoir un ou plusieurs ports. Le SM 1278 4xIO-Link Master a quatre ports. Un appareil IO-Link ou un capteur/actionneur standard peut être connecté à chaque port. IO-Link est un système de communication point-à-point.

12.3

Après la mise sous tension

Lors de la mise sous tension, l'appareil IO-Link est toujours dans l'état SIO (état I/O standard). Les ports du maître peuvent avoir différentes configurations. Reportez-vous au chapitre *IO-Link dans le Manuel système* de l'automate programmable *S7-1200* pour avoir plus de précisions.

Si un port est défini sur l'état SIO, le maître agit sur ce port comme une entrée numérique normale. Si le port est défini sur l'état IO-Link (mode de communication), le maître essaie de trouver l'appareil IO-Link connecté. Ce processus s'appelle la sortie de veille.

Pendant la sortie de veille, le maître envoie un signal défini et attend que l'esclave réponde. Au début, le maître tente de faire cette opération avec la vitesse en bauds la plus élevée possible. En cas d'échec, le maître essaie la vitesse en bauds plus faible suivante. Le maître essaie de communiquer avec l'appareil trois fois avec chaque vitesse en bauds. L'appareil prend toujours en charge uniquement une vitesse en bauds définie. Si le maître reçoit une réponse (autrement dit, si l'appareil est sorti de veille), les deux commenceront la communication. D'abord, ils échangent leurs paramètres de communication, puis ils commencent leur échange cyclique de données de traitement.

Si l'appareil esclave est retiré pendant le fonctionnement, le maître détecte l'abandon de communication, le signale avec la spécificité du bus de terrain au régulateur, et tente cycliquement de refaire sortir de veille l'appareil. Après une autre sortie de veille réussie, les paramètres de communication sont lus à nouveau, validés le cas échéant, puis le canal de communication cyclique redémarre.

12.4 protocole IO-Link

Le système IO-Link peut échanger trois types de données :

- données de traitement cyclique (entrées de données de traitement, sorties) → Données cycliques
- Paramètres de l'appareil (objets de données sur demande) → Données acycliques
- Événements → Données acycliques

L'appareil IO-Link n'envoie des données qu'après avoir reçu une demande du maître IO-Link de le faire. Les données de traitement sont envoyées après la trame IDLE du maître, et après que le maître demande expressément les données et les événements des paramètres de l'appareil.

12.5 Configuration dans le bus de terrain

Le maître IO-Link apparaît sur le bus de terrain comme un nœud de bus de terrain normal et est intégré via la description d'appareil appropriée dans le configurateur de réseau pertinent. Ces fichiers décrivent les propriétés de communication et d'autres propriétés du maître IO-Link, telles que le nombre de ports. Ils n'indiquent pas quels appareils IO-Link sont connectés.

Toutefois, la Description d'Appareil IO-Link (IODE) a été définie pour une représentation transparente complète de l'architecture du système jusqu'à l'appareil IO-Link. Avec l'aide de l'IODE et de l'outil de configuration IO-LinkS7-PCT, vous pouvez configurer quel appareil IO-Link est connecté à quel port de votre maître IO-Link.

Reportez-vous au système d'aide de S7-PCT et au *S7-1200 Programmable Controller System Manual* pour obtenir des informations détaillées sur la configuration.

12.6 IO-Link et le programme STEP 7

La communication acyclique des programmes maîtres IO-Link avec l'appareil IO-Link à l'aide du bloc fonctionnel (FB) IOL_CALL dans le programme du régulateur STEP 7 S7-1200. Le FB IOL_CALL indique le maître IO-Link que le programme utilise et les ports que le maître utilise pour l'échange de données.

Consultez le site Siemens Industry Online Support (<http://support.automation.siemens.com>) pour plus d'informations sur l'utilisation du FB IOL_CALL. Tapez "IO-Link" dans la zone de recherche pour consulter les informations relatives aux produits IO-Link et à leur emploi.

12.7 Le maître SM 1278 4xIO-Link

Le SM 1278 4xIO-Link Master est un module à 4 ports qui fonctionne à la fois comme un module de signaux et comme un module de communication. Chaque port peut fonctionner en mode IO-Link, entrée TOR 24 V CC ou sortie TOR 24 V CC uniques. Vous pouvez connecter jusqu'à quatre appareils IO-Link (connexion à 3 fils) ou quatre actionneurs standard ou codeurs standard.

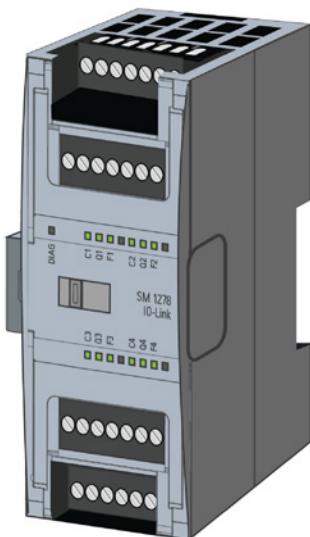
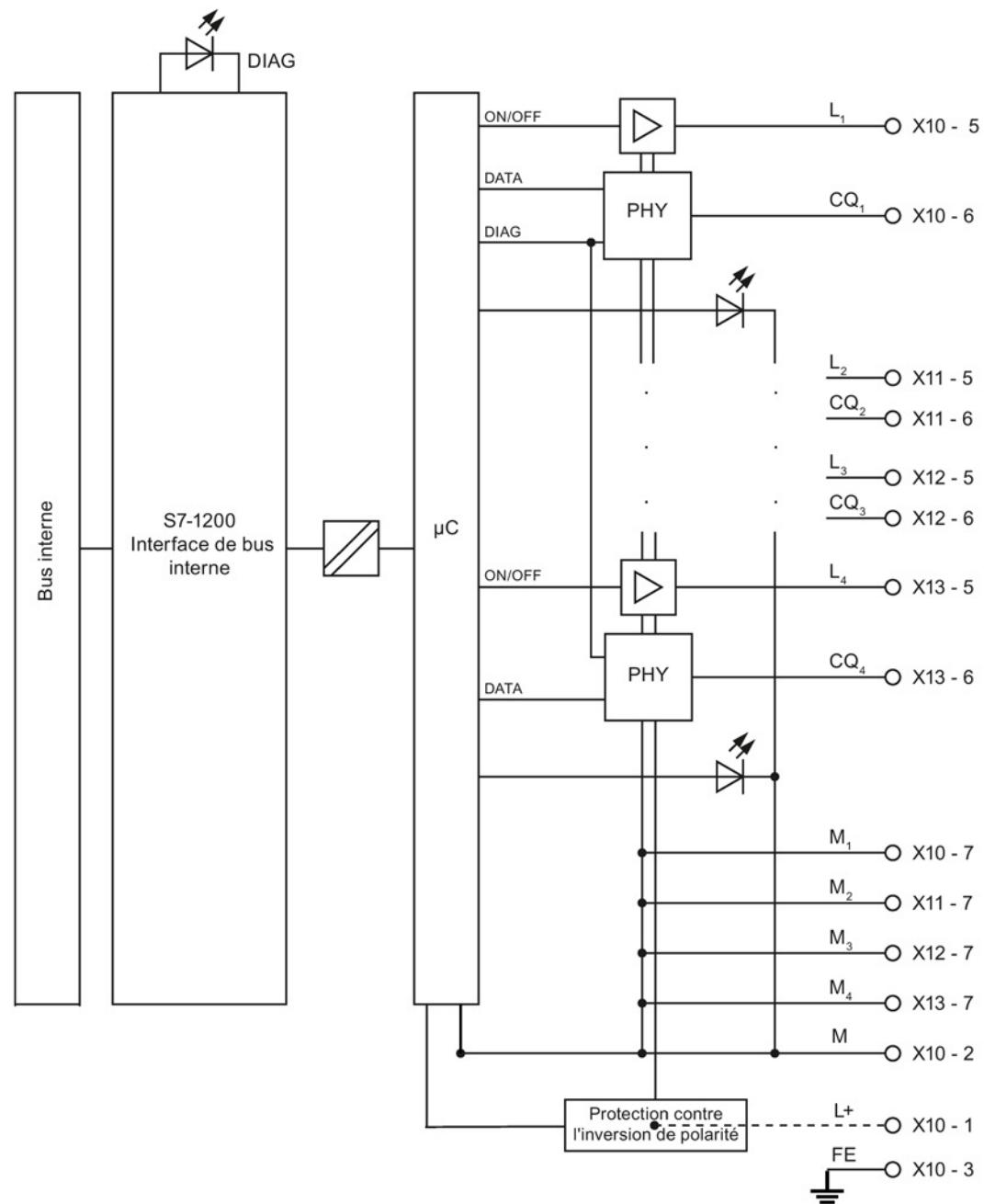
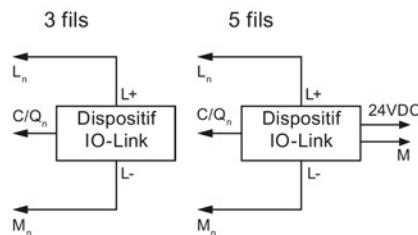


schéma fonctionnel Maître SM 1278 4xIO-Link

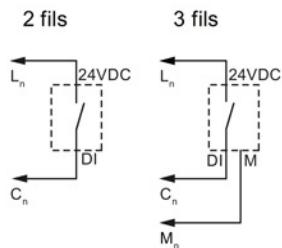


Exemples de raccordements

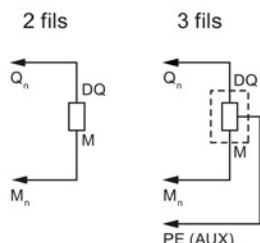
L'illustration suivante montre la configuration pour l'état de fonctionnement IO-Link (à 3 fils et à 5 fils), où n = nombre de ports :



L'illustration suivante montre la configuration pour l'état de fonctionnement DI (à 2 fils et à 3 fils), où n = nombre de ports :



L'illustration suivante montre la configuration pour l'état de fonctionnement DQ (à 2 fils et à 3 fils), où n = nombre de ports :



Informations détaillées sur l'utilisation et la configuration du Maître SM 1278 4xIO-Link

Pour des informations détaillées sur le SM 1278 4xIO-Link Master, comprenant des schémas, le raccordement, le paramétrage, les alarmes de diagnostic et plus, reportez-vous au Manuel système de l'automate programmable *S7-1200*.

Caractéristiques techniques

A.1 Caractéristiques techniques d'ordre général

Conformité aux normes

La conception du système d'automatisation S7-1200 est conforme aux normes et spécifications d'essai suivantes. Les critères de test pour le système d'automatisation S7-1200 se fondent sur ces normes et spécifications d'essai.

Notez que les modèles S7-1200 ne sont peut-être pas tous certifiés pour ces normes et que l'état d'homologation peut changer sans avis. Il est de votre responsabilité de déterminer les homologations applicables en se reportant aux marquages figurant sur le produit. Veuillez contacter votre agence Siemens si vous avez besoin d'informations supplémentaires concernant la dernière liste d'homologations exactes par numéro de référence.

homologation CE



Le système d'automatisation S7-1200 satisfait aux exigences et objectifs en matière de sécurité des directives CE énumérées ci-dessous et est conforme aux normes européennes harmonisées (EN) pour les automates programmables énumérées dans les Journaux officiels de l'Union Européenne.

- Directive CE 2006/95/CE (Basse tension) "Matériel électrique destiné à être employé dans certaines limites de tension"
 - EN 61131-2:2007 Automates programmables Spécifications et essais des équipements
- Directive CE 2004/108/CE (directive CEM) "Compatibilité électromagnétique"
 - Norme sur les émissions
EN 61000-6-4:2007+A1:2011 : environnement industriel
 - Norme sur l'immunité
EN 61000-6-2:2005 : environnement industriel
- Directive CE 94/9/CE (ATEX) "Appareils et systèmes de protection destinés à être utilisés en atmosphères explosives"
 - EN 60079-15:2010 : Classe de protection "n"

La déclaration de conformité CE est tenue à disposition des autorités compétentes auprès de :

Siemens AG
Sector Industry
I IA AS FA DH AMB
Postfach 1963
D-92209 Amberg
Germany

Homologation cULus



Underwriters Laboratories Inc. conformément à :

- Underwriters Laboratories, Inc. : listé UL 508 (Industrial Control Equipment)
- Canadian Standards Association : CSA C22.2 numéro 142 (Process Control Equipment)

Remarque

La gamme SIMATIC S7-1200 satisfait à la norme CSA.

La marque cULus indique que le S71200 a été examiné et certifié par les laboratoires Underwriters Laboratories (UL) comme étant conforme aux normes UL 508 et CSA 22.2 numéro 142.

Homologation FM



Factory Mutual Research (FM)

Norme d'homologation classe numéro 3600 et 3611

Homologué pour utilisation dans :

Classe I, Division 2, Groupe gaz A, B, C, D, Classe de température T3C Ta = 60 °C

Classe I, Zone 2, IIC, Classe de température T3 Ta = 60 °C

Classe canadienne I, Installation zone 2 par CEC 18-150

EXCEPTION IMPORTANTE : Voir les caractéristiques techniques pour le nombre d'entrées ou de sorties pouvant être à 1 simultanément. Certains modèles sont déclassés pour Ta = 60 °C.

ATTENTION

Le remplacement de composants peut remettre en cause la conformité à la classe I, division 2 et zone 2.

Les réparations ne doivent être effectuées que par un centre de service Siemens agréé.

Homologation IECEEx

EN 60079-0 : Atmosphères explosives - Règles générales

EN60079-15 : Matériel électrique pour atmosphères explosives ;

Type de protection 'nA'

IECEEx FMG14.0012X

Ex nA IIC Tx Gc

Des informations sur l'indice IECEEx peuvent apparaître sur le produit avec les informations sur les zones dangereuses FM.

Seuls les produits dotés d'un indice IECEEx sont homologués. Veuillez contacter votre agence Siemens si vous avez besoin d'informations supplémentaires concernant la dernière liste d'homologations exactes par numéro de référence.

Les modèles de relais ne sont pas inclus dans les homologations IECEEx.

Veuillez consulter le marquage spécifique du produit pour l'indice de température.

Installez les modules dans une enceinte adaptée fournissant un indice de protection minimum d'IP 54 conformément à la norme CEI 60079-15.

Homologation ATEX



L'homologation ATEX ne s'applique qu'aux modèles à courant continu. Elle ne s'applique pas aux modèles à courant alternatif et relais.

EN 60079-0:2009 : Atmosphères explosives - Règles générales

EN 60079-15:2010 : Matériel électrique pour atmosphères explosives ;

Type de protection 'nA'

II 3 G Ex nA II T4 ou T3 Gc

Installez les modules dans une enceinte adaptée procurant un niveau minimum de protection de IP54 conformément à la norme EN 60529, ou dans un emplacement procurant un niveau de protection équivalent.

Les câbles et les conducteurs associés doivent être classés pour la température réelle mesurée dans des conditions nominales.

L'installation doit garantir que les pointes de tension soient limitées à moins de 119 V. Reportez-vous à Immunité aux pointes de tension dans ce paragraphe.

EXCEPTION IMPORTANTE : Voir les caractéristiques techniques pour le nombre d'entrées ou de sorties pouvant être à 1 simultanément. Certains modèles sont déclassés pour $T_a = 60^\circ C$.

Homologation C-Tick



Le système d'automatisation S7-1200 satisfait aux exigences de la norme AS/NZS CISPR16 (classe A).

Caractéristiques techniques

A.1 Caractéristiques techniques d'ordre général

Certification coréenne



Le système d'automatisation S7-1200 satisfait aux exigences de l'homologation coréenne (marque KC). Il a été défini comme équipement de classe A et est conçu pour des applications industrielles et non pour un usage domestique.

Homologation de l'Union eurasienne des douanes (Biélorussie, Kazakhstan, Fédération de Russie)



EAC (Conformité eurasienne) : Déclaration de conformité selon la Norme technique de l'Union des douanes (TR CU)

Homologation pour le domaine maritime

Les produits S7-1200 sont régulièrement soumis à des homologations d'agences spéciales pour des marchés et des applications spécifiques. Veuillez contacter votre agence Siemens si vous avez besoin d'informations supplémentaires concernant la dernière liste d'homologations exactes par numéro de référence.

Sociétés de classification :

- ABS (American Bureau of Shipping)
- BV (Bureau Veritas)
- DNV (Det Norske Veritas)
- GL (Germanischer Lloyd)
- LRS (Lloyds Register of Shipping)
- Class NK (Nippon Kaiji Kyokai)
- Registre coréen de la navigation

Environnements industriels

Le système d'automatisation S7-1200 est conçu pour être utilisé dans des environnements industriels.

Tableau A- 1 Environnements industriels

Champ d'application	Exigences en matière d'émissions	Exigences en matière d'immunité	Exigences concernant l'immunité aux bruits
Industriel	EN 61000-6-4:2007+A1:2011	EN 61000-6-2:2005	EN 61000-6-2:2005

Remarque

Le système d'automatisation S7-1200 est destiné à une utilisation dans des zones industrielles ; une utilisation dans des zones résidentielles peut avoir un impact sur les réceptions radio ou TV. Si vous utilisez le S7-1200 dans des zones résidentielles, vous devez veiller à ce que son émission d'interférences radio respecte la valeur limite de la Classe B conformément à l'EN 55011.

Les exemples de mesures appropriées pour atteindre le niveau d'interférences RF, Classe B, comprennent :

- L'installation du S7-1200 dans une armoire de commande mise à la terre
- L'utilisation de filtres antiparasites dans les conduites d'alimentation

Veillez à ce que l'émission d'interférences radio respecte la Classe B conformément à l'EN 55011.

Une acceptation individuelle est requise (l'installation finale doit remplir toutes les conditions de sécurité de CEM d'une installation résidentielle).

Compatibilité électromagnétique

La compatibilité électromagnétique (CEM) est la capacité d'un appareil électrique à fonctionner comme prévu dans un environnement électromagnétique et à fonctionner sans émettre des niveaux d'interférence électromagnétique pouvant perturber d'autres appareils électriques à proximité.

Tableau A- 2 Immunité selon EN 61000-6-2

Compatibilité électromagnétique - Immunité selon EN 61000-6-2	
EN 61000-4-2 Décharge électrostatique	8 kV : décharge dans l'air vers toutes les surfaces 6 kV : décharge au contact vers les surfaces conductrices exposées
EN 61000-4-3 Essai d'immunité aux champs électromagnétiques rayonnés aux fréquences radioélectriques	80 à 1000 MHz, 10 V/m, 80% de modulation d'amplitude à 1 kHz 1,4 à 2,0 GHz, 3 V/m, 80% de modulation d'amplitude à 1 kHz 2,0 à 2,7 GHz, 1 V/m, 80% de modulation d'amplitude à 1 kHz
EN 61000-4-4 Salves transitoires rapides	2 kV, 5 kHz avec réseau de couplage vers système CC et CA 2 kV, 5 kHz avec blocage de couplage vers E/S
EN 61000-4-5 Immunité aux pointes de tension	Systèmes CA - état courant 2 kV, état différentiel 1 kV systèmes CC - état courant 2 kV, état différentiel 1 kV Pour des systèmes CC, reportez-vous à l'immunité aux pointes de tensions ci-dessous
EN 61000-4-6 Perturbations par conduction	150 kHz à 80 MHz, 10 V eff., 80 % de modulation d'amplitude à 1 kHz
EN 61000-4-11 Baisses de tension	Systèmes CA 0% pour 1 cycle, 40% pour 12 cycles et 70% pour 30 cycles à 60 Hz

Caractéristiques techniques

A.1 Caractéristiques techniques d'ordre général

Immunité aux pointes de tension

Les systèmes de câblage soumis à des pointes de tension provenant d'un couplage en cas de foudre doivent être équipés d'une protection externe. On trouve une caractéristique technique pour l'évaluation d'une protection des pointes de tension de type foudre dans EN 61000-4-5, avec des limites opérationnelles établies par EN 61000-6-2. Les CPU du S7-1200 DC et les modules entrées-sorties nécessitent une protection externe pour conserver un fonctionnement sécurisé lorsqu'ils sont soumis à des tensions de pointes de tension définies par cette norme.

Certains appareils qui prennent en charge la protection nécessaire d'immunité aux pointes de tension sont répertoriés ci-après. Ces appareils n'assurent la protection que s'ils sont convenablement installés selon les recommandations du fabricant. Les appareils fabriqués par d'autres fournisseurs ayant les mêmes caractéristiques techniques ou de meilleures caractéristiques peuvent également être utilisés :

Tableau A- 3 Appareils prenant en charge la protection d'immunité aux pointes de tension

Sous-système	Appareil de protection
alimentation +24 V CC	BLITZDUCTOR VT, BVT AVD 24, référence 918 422
Industrial Ethernet	DEHNpatch DPA M CLE RJ45B 48, référence 929 121
RS-485	BLITZDUCTOR XT, Basic Unit BXT BAS, référence 920 300
	BLITZDUCTOR XT, Module BXT ML2 BD HFS 5, référence 920 271
RS-232	BLITZDUCTOR XT, Basic Unit BXT BAS, référence 920 300
	BLITZDUCTOR XT, Module BXT ML2 BE S 12, référence 920 222
Entrées TOR +24 V CC	DEHN, Inc., Type DCO SD2 E 24, référence 917 988
Sorties TOR et alimentation capteur +24 V CC	DEHN, Inc., Type DCO SD2 E 24, référence 917 988
E/S analogiques	DEHN, Inc., Type DCO SD2 E 12, référence 917 987
Sorties relais	Aucune nécessaire

Tableau A- 4 Emissions conduites et rayonnées selon EN 61000-6-4

Compatibilité électromagnétique - Emissions conduites et rayonnées selon EN 61000-6-4		
Emissions conduites EN 55011, classe A, groupe 1	0,15 MHz à 0,5 MHz	<79dB (µV) quasi-pointe ; <66 dB (µV) moyenne
	0,5 MHz à 5 MHz	<73dB (µV) quasi-pointe ; <60 dB (µV) moyenne
	5 MHz à 30 MHz	<73dB (µV) quasi-pointe ; <60 dB (µV) moyenne
Emissions rayonnées EN 55011, classe A, groupe 1	30 MHz à 230 MHz	<40dB (µV/m) quasi-pointe, mesuré à 10 m
	230 MHz à 1 GHz	<47dB (µV/m) quasi-pointe, mesuré à 10 m
	1 GHz à 3 GHz	< 76dB (uV/m) quasi-pointe, mesuré à 10 m

Conditions ambiantes

Tableau A- 5 Transport et stockage

Conditions ambiantes - Transport et stockage	
EN 6006822, test Bb, chaleur sèche	- 40 °C à + 70 °C
EN 6006821, test Ab, froid	
EN 60068-2-30, test Db, chaleur humide saturée	25 °C à 55 °C, 95 % d'humidité
EN 60068-2-14, test Na, choc de température	- 40 °C à + 70 °C, temps de maintien 3 heures, 5 cycles
EN 60068-2-32, chute libre	0,3 m, 5 fois, emballage du produit
Pression atmosphérique	1080 à 660 hPa (correspond à une altitude de -1000 à 3500 m)

Tableau A- 6 Conditions de service

Conditions ambiantes - Fonctionnement	
Plage de température ambiante (admission d'air 25 mm en dessous de l'unité)	- 20 °C à 60 °C en montage horizontal - 20 °C à 50 °C en montage vertical 95 % d'humidité sans condensation Sauf indication contraire
Pression atmosphérique	1080 à 795 hPa (correspond à une altitude de -1000 à 2000 m)
Concentration de contaminants	SO ₂ : < 0,5 ppm ; H ₂ S : < 0,1 ppm ; RH < 60 % sans condensation ISA-S71.04 niveau de gravité G1, G2, G3
EN 60068-2-14, test Nb, changement de température	5 °C à 55 °C, 3 °C/minute
EN 60068-2-27 Choc mécanique	15 G, impulsion de 11 ms, 6 chocs dans chacun des 3 axes
EN 60068-2-6 Vibrations sinusoïdales	Montage sur profilé support : 3,5 mm de 5 à 9 Hz, 1G de 9 à 150 Hz Montage sur panneau : 7,0 mm de 5 à 9 Hz, 2G de 9 à 150 Hz 10 balayages par axe, 1 octave/minute

Caractéristiques techniques

A.1 Caractéristiques techniques d'ordre général

Tableau A- 7 Essai d'isolation à potentiel élevé

Essai d'isolation à potentiel élevé	
Circuits nominaux 24 V CC / 5 V CC	520 V CC (essai de type de limites d'isolation optique)
Circuits à la terre 115 V CA / 230 V CA	1500 V CA
Circuits 115 V CA / 230 V CA vers circuits 115 V CA / 230 V CA	1500 V CA
Circuits 115 V CA / 230 V CA vers circuits 24 V CC / 5 V CC	1 500 V CA (essai de type 3 000 V CA/4 242 V CC)
Port Ethernet à circuits 24 V CC / 5 V CC et terre ¹	1500 V CA (essai de type uniquement)

¹ L'isolation du port Ethernet est conçue pour limiter les risques pendant les pannes de réseau de courte durée avec tensions dangereuses. Elle ne satisfait pas aux exigences en matière de sécurité pour l'isolation des lignes CA de routine.

Classe de protection

- Classe de protection II selon EN 61131-2 (un conducteur de protection n'est pas requis)

Degré de protection

- IP20 Protection mécanique, EN 60529
- Protection contre le contact des doigts avec la haute tension comme testé par sonde standard. Une protection externe est nécessaire contre la poussière, la saleté, l'eau et les objets étrangers de diamètre inférieur à 12,5 mm.

Tensions nominales

Tableau A- 8 Tensions nominales

Tension nominale	Tolérance
24 V-	20,4 V- à 28,8 V-
120/230 V CA	85 V CA à 264 V CA, 47 à 63 Hz

Protection contre la tension inverse

Un circuit de protection contre la tension inverse est fourni sur chaque paire de bornes de l'alimentation + 24 V CC ou de l'alimentation d'entrée utilisateur pour les CPU, les modules d'entrées-sorties (SM) et les Signal Boards (SB). Endommager le système reste possible si l'on câble des paires de bornes différentes dans des polarités inverses.

Certains ports d'entrée d'alimentation 24 V CC dans le système S7-1200 sont interconnectés, avec un circuit logique commun connectant plusieurs bornes M. Par exemple, les circuits suivants sont interconnectés lorsqu'ils sont signalés comme "non isolés" dans les fiches techniques : l'alimentation 24 VCC de la CPU, la puissance du capteur de la CPU, l'entrée d'alimentation pour la bobine de relais d'un SM et l'alimentation pour une entrée analogique non isolée. Toutes les bornes M non isolées doivent être connectées au même potentiel de référence externe.

 ATTENTION
Connecter des bornes M non isolées à des potentiels de référence différents provoque des flux de courant indésirables qui peuvent être à l'origine de dégâts ou d'un fonctionnement imprévisible dans l'automate et tout équipement connecté.
Le non-respect de ces conseils peut être à l'origine de dégâts ou d'un fonctionnement imprévisible pouvant entraîner la mort, des blessures graves et des dommages matériels importants.
Vérifiez toujours que toutes les bornes M non isolées dans un système S7-1200 sont connectées au même potentiel de référence.

Sorties CC

Aucun circuit de protection contre les courts-circuits n'est fourni pour les sorties CC des CPU, modules d'entrées-sorties (SM) et Signal Boards (SB).

Durée d'utilisation électrique des relais

Les données de performance types estimées à partir des exemples de tests sont représentées ci-dessous. Les performances effectives peuvent varier selon votre application spécifique. Un circuit de protection externe qui est adapté à la charge améliorera la durée de vie des contacts. Les contacts N.C. ont une durée de vie type d'environ un tiers de celle du contact N.O. dans des conditions de charge inductive et de lampe.

Un circuit protecteur externe augmentera la durée de vie des contacts.

Tableau A- 9 Données de performance types

Données pour sélectionner un actionneur				
Courant thermique continu		2 A max.		
Commutation de la capacité et de la durée de vie des contacts				
	Pour une charge ohmique	Tension	Courant	Nombre de cycles de fonctionnement (type)
		24 V-	2,0 A	0,1 million

Caractéristiques techniques

A.1 Caractéristiques techniques d'ordre général

Données pour sélectionner un actionneur				
		24 V-	1,0 A	0,2 million
		24 V-	0,5 A	1,0 million
		48 V CA	1,5 A	1,5 million
		60 V CA	1,5 A	1,5 million
		120 V CA	2,0 A	1,0 million
		120 V CA	1,0 A	1,5 million
		120 V CA	0,5 A	2,0 millions
		230 V CC	2,0 A	1,0 million
		230 V CC	1,0 A	1,5 million
		230 V CC	0,5 A	2,0 millions
	Pour une charge inductive (selon IEC 947-5-1 DC13/AC15)	Tension	Courant	Nombre de cycles de fonctionnement (type)
		24 V-	2,0 A	0,05 million
		24 V-	1,0 A	0,1 million
		24 V-	0,5 A	0,5 million
		24 V CA	1,5 A	1,0 million
		48 V CA	1,5 A	1,0 million
		60 V CA	1,5 A	1,0 million
		120 V CA	2,0 A	0,7 million
		120 V CA	1,0 A	1,0 million
		120 V CA	0,5 A	1,5 million
		230 V CC	2,0 A	0,7 million
		230 V CC	1,0 A	1,0 million
		230 V CC	0,5 A	1,5 million
	Activation d'une entrée numérique	Possible		
	Fréquence de commutation			
	Mécanique	max. 10 Hz		
	Pour une charge ohmique	max. 1 Hz		
	Pour une charge inductive (selon IEC 947-5-1 DC13/AC15)	max. 0,5 Hz		
	Pour une charge de lampe	max. 1Hz		

Rémanence de la mémoire interne de la CPU

- Durée de vie des données rémanentes et des données des journaux de données : 10 ans
- Données rémanentes hors tension, durée des écritures en cycles : 2 million de cycles
- Données des journaux de données, jusqu'à 2 Ko par entrée de journal de données, durée des écritures en cycles : 500 millions d'entrées dans le journal de données

Remarque

Effet des journaux de données sur la mémoire interne de la CPU

Chaque écriture dans un journal de données utilise au minimum 2 Ko de mémoire. Si votre programme écrit fréquemment de petites quantités de données, il utilise au moins 2 Ko de mémoire à chaque écriture. Une meilleure solution consiste à accumuler les petits éléments de données dans un bloc de données (DB) et à écrire le bloc de données dans le journal de données à des intervalles moins fréquents.

Si votre programme écrit de nombreuses entrées dans le journal de données à une fréquence élevée, vous devriez envisager d'utiliser une carte mémoire SD remplaçable.

A.2 Modules CPU

Pour une liste plus complète des modules disponibles pour le S7-1200, reportez-vous au *Manuel Système de l'automate programmable S7-1200* ou au site web du service client (<http://www.siemens.com/tiaprofile>).

Tableau A- 10Caractéristiques générales

Caractéristiques générales		CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C
Référence	Relais CA/CC	6ES7 211-1BE40-0XB0	6ES7 212-1BE40-0XB0	6ES7 214-1BG40-0XB0	6ES7 215-1BG40-0XB0	--
	DC/DC/Relay	6ES7 211-1HE40-0XB0	6ES7 212-1HE40-0XB0	6ES7 214-1HG40-0XB0	6ES7 215-1HG40-0XB0	--
	CC/CC/CC	6ES7 211-1AE40-0XB0	6ES7 212-1AE40-0XB0	6ES7 214-1AG40-0XB0	6ES7 215-1AG40-0XB0	6ES7 217-1AG40-0XB0
Dimensions L x H x P (mm)		90 x 100 x 75		110 x 100 x 75	130 x 100 x 75	150 x 100 x 75
Poids	<ul style="list-style-type: none"> AC/DC/RLY DC/DC/Relay DC/DC/DC 	<ul style="list-style-type: none"> 420 grammes 380 grammes 370 grammes 	<ul style="list-style-type: none"> 425 grammes 385 grammes 370 grammes 	<ul style="list-style-type: none"> 475 grammes 435 grammes 415 grammes 	<ul style="list-style-type: none"> 585 grammes 550 grammes 520 grammes 	<ul style="list-style-type: none"> - - 530 grammes
Puissance dissipée	<ul style="list-style-type: none"> AC/DC/RLY DC/DC/Relay DC/DC/DC 	<ul style="list-style-type: none"> 10 W 8 W 8 W 	<ul style="list-style-type: none"> 11 W 9 W 9 W 	<ul style="list-style-type: none"> 14 W 12 W 12 W 	<ul style="list-style-type: none"> 14 W 12 W 12 W 	<ul style="list-style-type: none"> - - 12 W
Courant disponible (5 V-) pour bus SM et CM		750 mA max.	1000 mA max.	1600 mA max.	1600 mA max.	1600 mA max.
Courant disponible (24 V-) alim. capteur		300 mA max.	300 mA max.	400 mA max.	400 mA max.	400 mA max.
Consommation de courant entrées TOR (24 V-)		4 mA/entrée utilisée	4 mA/entrée utilisée	4 mA/entrée utilisée	4 mA/entrées utilisées	4 mA/entrée utilisée

Tableau A- 11Caractéristiques de la CPU

Caractéristiques de la CPU	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C	
Mémoire utilisateur						
• Mémoire de travail	• 50 Ko	• 75 Ko	• 100 Ko	• 125 Ko	• 150 Ko	
• Mémoire de chargement	• 1 Mo	• 1 Mo	• 4 Mo	• 4 Mo	• 4 Mo	
• Mémoire rémanente	• 10 Ko	• 10 Ko	• 10 Ko	• 10 Ko	• 10 Ko	
E/S TOR intégrées Voir caractéristiques (Page 405).	6 entrées 4 sorties	8 entrées 6 sorties	14 entrées 10 sorties	14 entrées 10 sorties	14 entrées 10 sorties	
E/S analogiques intégrées Voir caractéristiques (Page 416).	2 entrées	2 entrées	2 entrées	2 entrées 2 sorties	2 entrées 2 sorties	
Taille de la mémoire image						
• Entrées	• 1024 octets	• 1024 octets	• 1024 octets	• 1024 octets	• 1024 octets	
• Sorties	• 1024 octets	• 1024 octets	• 1024 octets	• 1024 octets	• 1024 octets	
Mémentos (M)	4096 octets	4096 octets	8192 octets	8192 octets	8192 octets	
Mémoire temporaire (locale)	<ul style="list-style-type: none"> 16 Ko pour le démarrage et le cycle du programme (FB et FC associés inclus) 6 Koctets pour chacun des autres niveaux de priorité d'interruption (y compris les FB et les FC) 					
extension SM	Aucun	2 SM max.	8 SM max.	8 SM max.	8 SM max.	
SB, CB ou BB pour extension	1 max.	1 max.	1 max.	1 max.	1 max.	
CM pour extension	3 max.	3 max.	3 max.	3 max.	3 max.	
Compteurs rapides	Total	Jusqu'à 6 configurés pour utiliser n'importe quelles entrées intégrées ou SB				
	1 MHz	--	--	--	--	lb.2 à lb.5 (Différentiel)
	100/180 kHz	la.0 à la.5	la.0 à la.5	la.0 à la.5	la.0 à la.5	la.0 à la.5
	30/120 kHz	--	la.6 à la.7	la.6 à lb.5	la.6 à lb.5	la.6 à lb.1
Sorties d'impulsions ²	Total	Jusqu'à 4 configurés pour utiliser n'importe quelles sorties intégrées ou SB				
	1 MHz	--	--	--	--	Qa.0 à Qa.3 (Différentiel)
	100 kHz	Qa.0 à Qa.3	Qa.0 à Qa.3	Qa.0 à Qa.3	Qa.0 à Qa.3	Qa.4 à Qb.1
	30 kHz	--	Qa.4 à Qa.5	Qa.4 à Qb.1	Qa.4 à Qb.1	--
Entrées de capture d'impulsions	6	8	14	14	14	
Alarmes temporisées	4 au total avec résolution de 1 ms	4 au total avec résolution de 1 ms	4 au total avec résolution de 1 ms	4 au total avec résolution de 1 ms	4 au total avec résolution de 1 ms	
Alarmes cycliques	4 au total avec résolution de 1 ms	4 au total avec résolution de 1 ms	4 au total avec résolution de 1 ms	4 au total avec résolution de 1 ms	4 au total avec résolution de 1 ms	

Caractéristiques de la CPU	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C
Alarmes sur front	6 pour front montant et 6 pour front descendant	8 pour front montant et 8 pour front descendant	12 pour front montant et 12 pour front descendant	12 pour front montant et 12 pour front descendant	12 pour front montant et 12 pour front descendant
Avec SB optionnel	10 pour front montant et 10 pour front descendant	12 pour front montant et 12 pour front descendant	16 pour front montant et 16 pour front descendant	16 pour front montant et 16 pour front descendant	16 pour front montant et 16 pour front descendant
Horloge temps réel	<ul style="list-style-type: none"> Précision Durée de rémanence (supercondensateur sans maintenance) 	<ul style="list-style-type: none"> +/- 60 secondes/mois 20 jours typ./12 jours min. à 40 °C 	<ul style="list-style-type: none"> +/- 60 secondes/mois 20 jours typ./12 jours min. à 40 °C 	<ul style="list-style-type: none"> +/- 60 secondes/mois 20 jours typ./12 jours min. à 40 °C 	<ul style="list-style-type: none"> +/- 60 secondes/mois 20 jours typ./12 jours min. à 40 °C
Vitesse d'exécution	<ul style="list-style-type: none"> BOOL Transférer mot Opérations arithmétiques sur réels 	<ul style="list-style-type: none"> 0,08 µs/instruction 1,7µs/ instruction 2,3 µs/instruction 	<ul style="list-style-type: none"> 0,08 µs/instruction 1,7µs/ instruction 2,3 µs/instruction 	<ul style="list-style-type: none"> 0,08 µs/instruction 1,7µs/ instruction 2,3 µs/instruction 	<ul style="list-style-type: none"> 0,08 µs/instruction 1,7 µs/instruction 2,3 µs/instruction

¹ La vitesse la plus faible s'applique quand le HSC est configuré pour l'état de fonctionnement en quadrature.

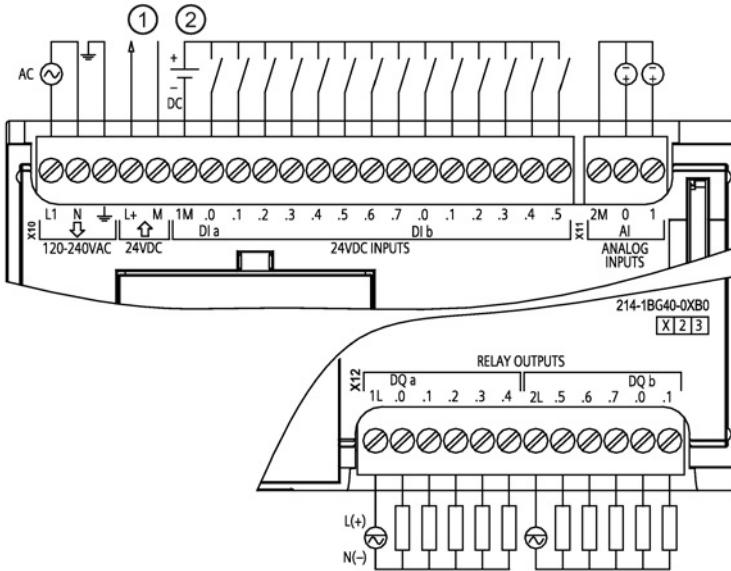
² Pour les modèles de CPU avec sorties relais, vous devez installer un Signal Board (SB) numérique pour utiliser les sorties d'impulsions.

Tableau A- 12Communication

Caractéristiques techniques	CPU 1211C, CPU 1212C, CPU 1214C	CPU 1215C, CPU 1217C
Communication	<p>1 port Ethernet</p> <ul style="list-style-type: none"> • Débits • Isolation (signal externe à logique API) • Type de câble 	<p>2 ports Ethernet</p> <ul style="list-style-type: none"> • 10/100 Mb/s • Isolé par transformateur, 1500 V- • CAT5e blindé
Appareils	<ul style="list-style-type: none"> • 4 HMI • 1 PG 	<ul style="list-style-type: none"> • 4 HMI • 1 PG
Liaisons Ethernet ¹	8 (actives ou passives)	8 (actives ou passives)
Liaisons CPU à CPU S7 (GET/PUT)	<ul style="list-style-type: none"> • 8 (client) • 3 (serveur) 	<ul style="list-style-type: none"> • 8 (client) • 3 (serveur)

¹ Liaisons pour la communication ouverte (Open User Communications) active ou passive : TSEND_C, TRCV_C, TCON, TDISCON, TSEND et TRCV.

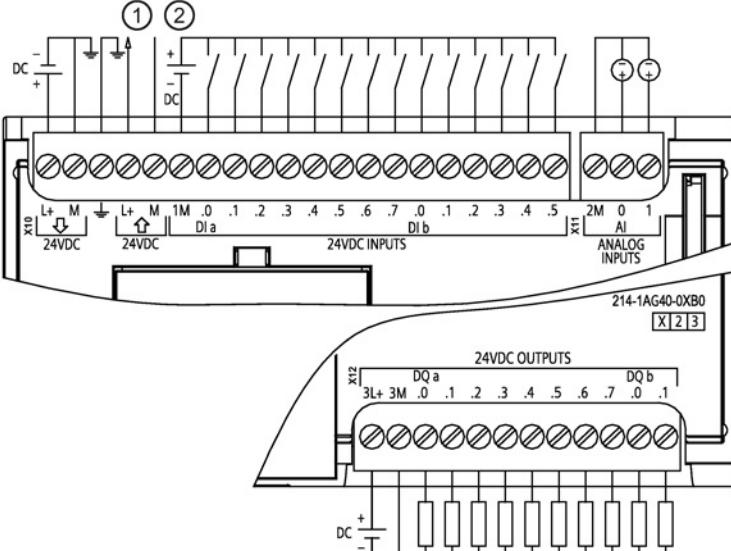
Tableau A- 13Schémas de câblage pour CPU 1214C AC/DC/Relais

CPU 1214C AC/DC/Relay
 <p>① Sortie alim. capteur 24 V CC. Pour plus d'immunité aux bruits, connectez "M" à la terre du châssis même si vous n'utilisez pas l'alimentation de capteur.</p> <p>② Pour les entrées de type P, connectez "-" à "M" (comme illustré). Pour les entrées de type N, connectez "+" à "M".</p> <p>Note 1 : les connecteurs X11 doivent être en or. Reportez-vous au <i>Manuel système de l'automate programmable S7-1200</i>, Annexe C, Pièces détachées pour le numéro de référence.</p> <p>Note 2 : Soit la borne L1, soit la borne N (L2) peuvent être connectées à une source de tension pouvant aller jusqu'à 240 V CA. La borne N peut être considérée L2 et ne nécessite pas d'être mise à la terre. Aucune polarisation n'est requise pour les bornes L1 et N (L2).</p> <p>Note 3 : Pour obtenir des informations sur le port Ethernet de la CPU, reportez-vous au <i>Manuel système de l'automate programmable S7-1200</i>, Configuration de l'appareil.</p>

Caractéristiques techniques

A.2 Modules CPU

Tableau A- 14Schémas de câblage pour CPU 1214C AC/DC/DC

CPU 1214C DC/DC/DC		
	<p>① Sortie alim. capteur 24 V CC. Pour plus d'immunité aux bruits, connectez "M" à la terre du châssis même si vous n'utilisez pas l'alimentation de capteur.</p> <p>② Pour les entrées de type P, connectez "-" à "M" (comme illustré). Pour les entrées de type N, connectez "+" à "M".</p> <p>Note 1 : les connecteurs X11 doivent être en or. Reportez-vous au <i>Manuel système de l'automate programmable S7-1200</i>, Annexe C, Pièces détachées pour le numéro de référence.</p> <p>Note 2 : Pour obtenir des informations sur le port Ethernet de la CPU, reportez-vous au <i>Manuel système de l'automate programmable S7-1200</i>, Configuration de l'appareil.</p>	

A.3 Modules E/S numériques

Pour une liste plus complète des modules disponibles pour le S7-1200, reportez-vous au *Manuel Système de l'automate programmable S7-1200* ou au site web du service client (<http://www.siemens.com/tiportal>).

A.3.1 Entrée/sortie TOR SB 1221 SB 1222 et SB 1223 (DI, DQ et DI/DQ)

Tableau A- 15Modules d'entrée numérique (DI) SB 1221 et de sortie numérique (DQ) SB 1222

Généralités		SB 1221 4 DI (200 kHz)	SB 1222 4 DQ (200 kHz)
Référence		<ul style="list-style-type: none"> 24 V- : 6ES7 221-3BD30-0XB0 5 V- : 6ES7 221-3AD30-0XB0 	<ul style="list-style-type: none"> 24 V- : 6ES7 222-1BD30-0XB0 5 V- : 6ES7 222-1AD30-0XB0
Dimensions L x H x P (mm)		38 x 62 x 21	38 x 62 x 21
Poids		35 grammes	35 grammes
Puissance dissipée		<ul style="list-style-type: none"> 24 V- : 1,5 W 5 V- : 1,0 W 	0,5 W
Consommation de courant	Bus SM	40 mA	35 mA
	24 V-	<ul style="list-style-type: none"> 24 V- : 7 mA / entrée + 20 mA 5 V- : 15 mA / entrée + 15 mA 	15 mA
Entrées/sorties		4 entrées (source)	4 sorties (état solide - MOFSET)

Tableau A- 16modules de combinaison entrée/sortie TOR (DI / DQ) SB 1223

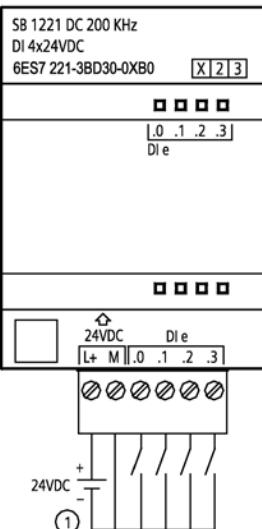
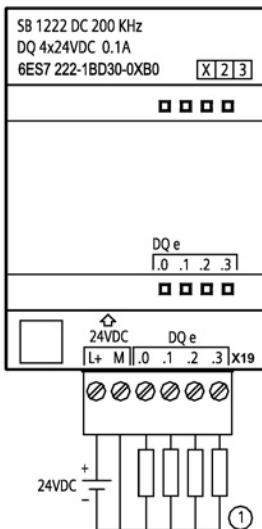
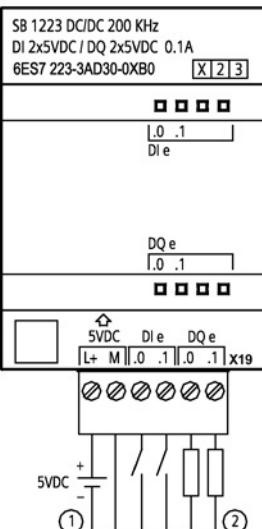
Général		DI / DQ SB 1223 (200 kHz)	2 DI / DQ SB 1223
Référence		<ul style="list-style-type: none"> 24 V- : 6ES7 223-3BD30-0XB0 5 V- : 6ES7 223-3AD30-0XB0 	24 V- : 6ES7 223-0BD30-0XB0
Dimensions L x H x P (mm)		38 x 62 x 21	38 x 62 x 21
Poids		35 grammes	40 grammes
Puissance dissipée		<ul style="list-style-type: none"> 24 V- : 1,0 W 5 V- : 0,5 W 	24 V- : 1,0 W
Consommation de courant	Bus SM	35 mA	50 mA
	24 V-	<ul style="list-style-type: none"> 24 V- : 7 mA / entrée + 20 mA 5 V- : 15 mA / entrée + 15 mA 	4 mA / entrée utilisée
Entrées/sorties		2 entrées (source) 2 sorties (état solide - MOFSET)	2 entrées (Type IEC 1 de type N) 2 sorties (état solide - MOFSET)

Remarque

Les SB rapides (200 kHz) utilisent des entrées "source". Le SB standard (20 kHz) utilise des entrées "sink". Reportez-vous aux caractéristiques pour les entrées et sorties TOR (Page 405).

Les sorties rapides (200 kHz) (SB 1222 et SB 1223) peuvent être soit "source", soit "sink". Pour les sorties de type P, connectez "Load" à "-" (comme sur la figure). Pour les sorties de type N, connectez "Load" à "+". Comme les deux configurations "sink" et "source" (P et N) sont prises en charge par le même ensemble de circuits, l'état actif d'une charge "source" est l'opposé de celui d'une charge "sink". Une sortie "source" fournit une logique positive (bit Q et DEL activés lorsque la charge présente un flux de courant) alors qu'une sortie "sink" fournit une logique négative (bit Q et DEL désactivés lorsque la charge présente un flux de courant). Si le module est enfiché sans programme utilisateur, le paramétrage par défaut pour ce module est 0V, ce qui signifie qu'une charge "sink" sera activée.

Tableau A- 17Schémas de câblage pour les SB TOR

Module d'entrée SB 1221	Module de sortie SB 1222	Module d'entrée/sortie SB 1223
<p>DI 4 SB 1221 (200 kHz)</p>  <p>SB 1221 DC 200 KHz DI 4x24VDC 6ES7 221-3BD30-0XB0 [X 2 3]</p> <p>□ □ □ [.0 .1 .2 .3] DI e</p> <p>□ □ □ 24VDC [L+ M] [.0 .1 .2 .3] DI e</p> <p>□ □ □ 24VDC [L+ M] [.0 .1 .2 .3]</p> <p>24VDC + / / / / -</p> <p>①</p>	<p>DQ 4 SB 1222 (200 kHz)</p>  <p>SB 1222 DC 200 KHz DQ 4x24VDC 0.1A 6ES7 222-1BD30-0XB0 [X 2 3]</p> <p>□ □ □ DQ e [.0 .1 .2 .3]</p> <p>□ □ □ 24VDC [L+ M] [.0 .1 .2 .3] x19 DQ e</p> <p>□ □ □ 24VDC [L+ M] [.0 .1 .2 .3]</p> <p>24VDC + / / / / -</p> <p>①</p>	<p>SB 1223 DI 2 / DQ 2 (200 kHz)</p>  <p>SB 1223 DC/DC 200 KHz DI 2x5VDC / DQ 2x5VDC 0.1A 6ES7 223-3AD30-0XB0 [X 2 3]</p> <p>□ □ □ [.0 .1] DI e</p> <p>□ □ □ DQ e [.0 .1]</p> <p>□ □ □ 5VDC [L+ M] [.0 .1] DI e</p> <p>□ □ □ 5VDC [L+ M] [.0 .1] x19 DQ e</p> <p>5VDC + / / / / -</p> <p>① ②</p>

① Prise en charge d'entrées de type P uniquement

① Pour les sorties de type P, connectez "Load" à "-" (comme sur la figure). Pour les sorties de type N, connectez "Load" à "+". Comme les deux configurations "sink" et "source" sont prises en charge par le même ensemble de circuits, l'état actif d'une charge "source" est l'opposé de celui d'une charge "sink". Une sortie "source" fournit une logique positive (bit Q et DEL activés lorsque la charge présente un flux de courant) alors qu'une sortie "sink" fournit une logique négative (bit Q et DEL désactivés lorsque la charge présente un flux de courant). Si le module est enfilé sans programme utilisateur, le paramétrage par défaut pour ce module est 0 V, ce qui signifie qu'une charge "sink" sera activée.

① Prise en charge d'entrées de type P uniquement

② Pour les sorties de type P, connectez "Load" à "-" (comme sur la figure). Pour les sorties de type N, connectez "Load" à "+". Comme les deux configurations "sink" et "source" (P et N) sont prises en charge par le même ensemble de circuits, l'état actif d'une charge "source" est l'opposé de celui d'une charge "sink". Une sortie "source" fournit une logique positive (bit Q et DEL activés lorsque la charge présente un flux de courant) alors qu'une sortie "sink" fournit une logique négative (bit Q et DEL désactivés lorsque la charge présente un flux de courant). Si le module est enfilé sans programme utilisateur, le paramétrage par défaut pour ce module est 0 V, ce qui signifie qu'une charge "sink" sera activée.

Remarque

Les SB rapides (200 kHz) (SB 1221 et SB 1223) prennent en charge uniquement des entrées "sink". Le SB 1223 standard prend en charge uniquement les entrées de type P.

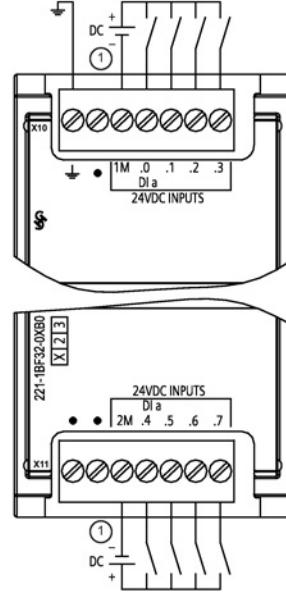
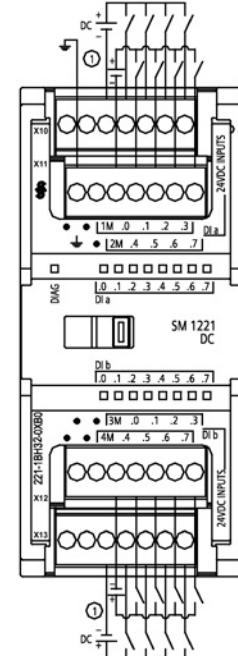
Les sorties rapides (200 kHz) (SB 1222 et SB 1223) peuvent être soit "source", soit "sink". Pour les sorties "source", connectez "Load" à "-" (comme sur la figure). Pour les sorties "sink", connectez "Load" à "+". Comme les deux configurations "sink" et "source" (P et N) sont prises en charge par le même ensemble de circuits, l'état actif d'une charge "source" est l'opposé de celui d'une charge "sink". Une sortie "source" fournit une logique positive (bit Q et DEL activés lorsque la charge présente un flux de courant) alors qu'une sortie "sink" fournit une logique négative (bit Q et DEL désactivés lorsque la charge présente un flux de courant). Si le module est enfiché sans programme utilisateur, le paramétrage par défaut pour ce module est 0 V, ce qui signifie qu'une charge "sink" sera activée.

A.3.2 Entrée numérique (DI) SM 1221

Tableau A- 18 Entrée numérique (DI) SM 1221

Caractéristiques techniques	SM 1221 DI 8 24 VDC	SM 1221 DI 16 24 VDC
Référence	6ES7 221-1BF32-0XB0	6ES7 221-1BH32-0XB0
Nombre d'entrées (DI)	8	16
Voir caractéristiques (Page 405).		
Dimensions L x H x P (mm)	45 x 100 x 75	45 x 100 x 75
Poids	170 grammes	210 grammes
Dissipation de courant	1,5 W	2,5 W
Consommation de courant	Bus SM 24 V-	105 mA 4 mA/ entrée utilisée
		130 mA 4 mA / entrée utilisée

Tableau A- 19 Schéma de câblage pour les modules d'entrée TOR (DI) SM 1221

SM 1221 DI 8 (24 VDC)	SM 1221 DI 16 (24 VDC)
	

① Pour les entrées de type N, connectez "-" à "M" (comme sur la figure). Pour les entrées de type P, connectez "+" à "M".

A.3.3 Sortie numérique (DQ) SM 1222

Tableau A- 20 Sortie numérique (DQ) SM 1222

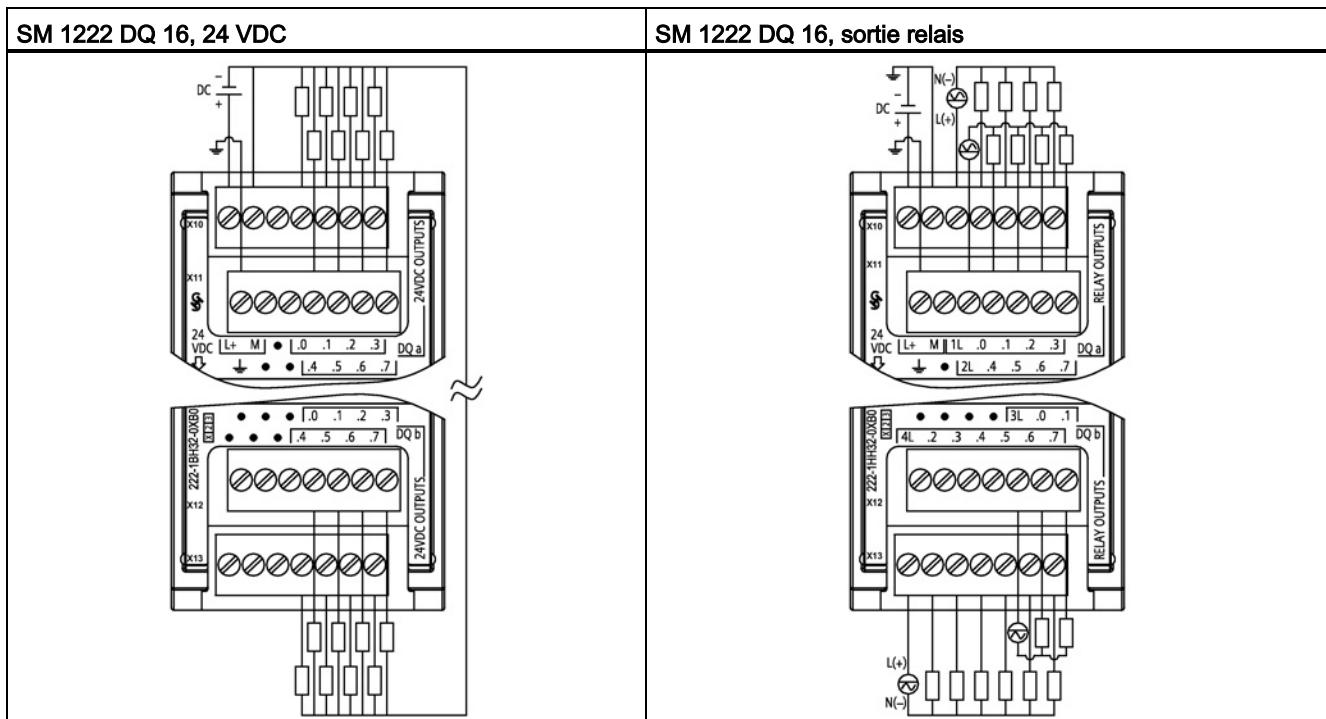
Caractéristiques techniques	DQ SM 1222 (Relais)	SM 1222 DQ (24 VDC)
Référence	<ul style="list-style-type: none"> DQ 8 : 6ES7 222-1HF32-0XB0 DQ 8 : inverseur : 6ES7 222-1XF32-0XB0 DQ 16 : 6ES7 222-1HH32-0XB0 	<ul style="list-style-type: none"> DQ 8 : 6ES7 222-1BF32-0XB0 DQ 16 : 6ES7 222-1BH32-0XB0
Nombre de sorties (DQ) Voir caractéristiques (Page 405).	<ul style="list-style-type: none"> 8 A (Commutation DQ 8 et DQ 8) 16 (DQ 16) 	<ul style="list-style-type: none"> 8 (DQ 8) 16 (DQ 16)
Dimensions L x H x P (mm)	<ul style="list-style-type: none"> DQ 8 et DQ 16 : 45 x 100 x 75 Commutation DQ 8 : 70 x 100 x 75 	45 x 100 x 75
Poids	<ul style="list-style-type: none"> DQ 8 : 190 grammes Commutation DQ 8 : 310 grammes DQ 16 : 260 grammes 	<ul style="list-style-type: none"> DQ 8 : 180 grammes DQ 16 : 220 grammes

Caractéristiques techniques

A.3 Modules E/S numériques

Caractéristiques techniques		DQ SM 1222 (Relais)	SM 1222 DQ (24 VDC)
Puissance dissipée		<ul style="list-style-type: none"> DQ 8 : 4,5 W Commutation DQ 8 : 5 W DQ 16 : 8,5 W 	<ul style="list-style-type: none"> DQ 8 : 1,5 W DQ 16 : 2,5 W
Consommation de courant	Bus SM	<ul style="list-style-type: none"> DQ 8 : 120 mA Commutation DQ 8 : 140 mA DQ 16 : 135 mA 	<ul style="list-style-type: none"> DQ 8 : 120 mA DQ 16 : 140 mA
	24 V-	<ul style="list-style-type: none"> DQ 8 et DQ 16:11 mA / Bobine de relais utilisée Commutation DQ 8 : 16,7 mA Bobine de relais utilisée 	<ul style="list-style-type: none"> DQ 8 : -- DQ 16 : --

Tableau A- 21Schéma de câblage pour les modules de sortie numérique (DQ) SM 1222

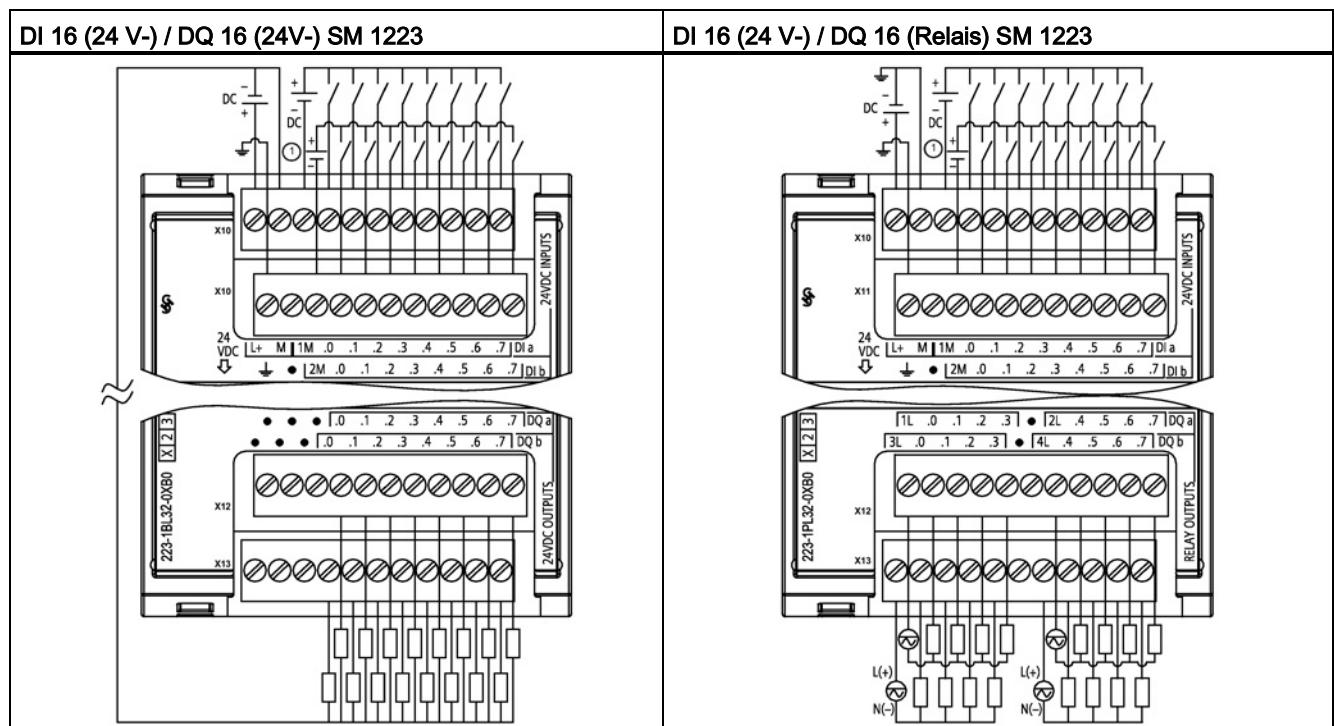


A.3.4 Entrées/sorties TOR (DI / DQ) SM 1223

Tableau A- 22Combinaison entrée/sortie TOR (DI / DQ) SM 1223

Caractéristiques techniques	SM 1223 DI (24 VDC) / DQ (Relay)	SM 1223 DI (24 VDC) / DQ (24 VDC)
Référence	DI 8 / DQ 8 : 6ES7 223-1PH32-0XB0 DI 16 / DQ 16 : 6ES7 223-1PL32-0XB0	DI 8 / DQ 8 : 6ES7 223-1BH32-0XB0 DI 8 / DQ 8 : 6ES7 223-1BL32-0XB0
Nombre d'entrées-sorties (DI / DQ) Voir caractéristiques (Page 405).	<ul style="list-style-type: none"> Entrées : 8 ou 16 (24 V-) Sorties : 8 ou 16 (relais) 	<ul style="list-style-type: none"> Entrées : 8 ou 16 (24 V-) Sorties : 8 ou 16 (24 V-)
Dimensions L x H x P (mm)	<ul style="list-style-type: none"> DI 8 / DQ 8 : 45 x 100 x 75 DI 16 / DQ 16 : 70 x 100 x 75 	<ul style="list-style-type: none"> DI 8 / DQ 8 : 45 x 100 x 75 DI 16 / DQ 16 : 70 x 100 x 75
Poids	<ul style="list-style-type: none"> DI 8 / DQ 8 : 230 grammes DI 16 / DQ 16 : 350 grammes 	<ul style="list-style-type: none"> DI 8 / DQ 8 : 210 grammes DI 16 / DQ 16 : 310 grammes
Puissance dissipée	<ul style="list-style-type: none"> DI 8 / DQ 8 : 5,5 W DI 16 / DQ 16 : 10 W 	<ul style="list-style-type: none"> DI 8 / DQ 8 : 2,5 W DI 16 / DQ 16 : 4,5 W
Consommation de courant	Bus SM	<ul style="list-style-type: none"> DI 8 / DQ 8 : 145 mA DI 16 / DQ 16 : 180 mA
	24 V-	4 mA / entrée utilisée 11 mA / bobine de relais utilisée
		4 mA/entrée utilisée

Tableau A- 23Schémas de câblage pour modules de combinaison DI / DQ SM 1223



① Pour les entrées de type N, connectez "-" à "M" (comme sur la figure). Pour les entrées de type P, connectez "+" à "M".

Caractéristiques techniques

A.3 Modules E/S numériques

A.3.5 SM 1223 120/230 VAC input / Relay output

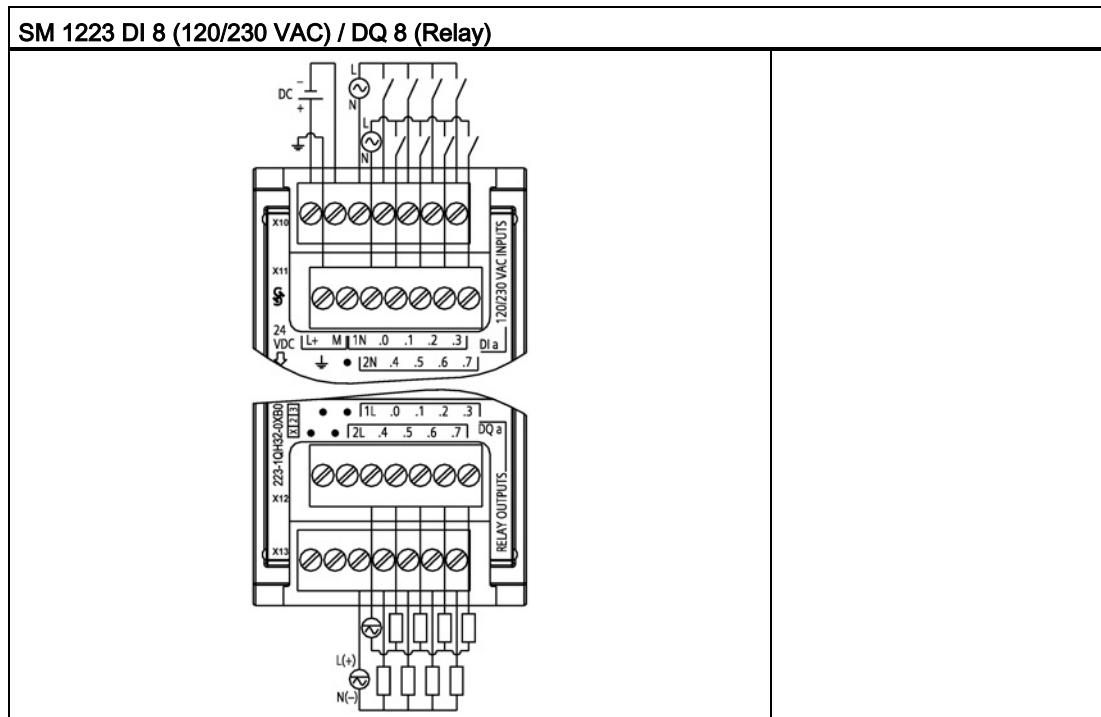
Tableau A- 24Combinaison entrées/sorties TOR (DI / DQ) SM 1223

Caractéristiques techniques	SM 1223 DI (120/230 VAC) / DQ (Relay)
Référence	DI 8 / DQ 8 : 6ES7 223-1QH32-0XB0
Nombre d'entrées-sorties (DI / DQ)	Entrées : 8 (120/230 VAC) Voir les caractéristiques techniques des entrées 120/230 VCA (Page 407). Sorties : 8 (relais) Voir les caractéristiques techniques des sorties TOR (Page 408).
Dimensions L x H x P (mm)	45 x 100 x 75
Poids	190 grammes
Puissance dissipée	7,5 W
Consommation de courant	Bus SM 24 V- 120 mA 11 mA / bobine de relais utilisée

Remarque

Le module d'entrées-sorties SM 1223 DI 8 x 120/230 VAC, DQ 8 x relais (6ES7 223-1QH32-0XB0) est homologué pour une utilisation en Classe 1, Division 2, Groupe gaz A, B, C, D, Classe de température T4 Ta = 40 °C.

Tableau A- 25Schéma de câblage du SM 1223 DI 8 (120/230 VAC) / DQ 8 (Relay)



A.4 Caractéristiques pour les entrées et sorties numériques

A.4.1 Entrées TOR (DI) 24 V CC

Tableau A- 26Caractéristiques des entrées numériques (DI)

Caractéristiques techniques	CPU, SM et SB	SB rapide (200 kHz)
Type	<ul style="list-style-type: none"> CPU et SM : CEI type 1 en mode P (P/N) SB 1223 : CEI type 1 en mode N (uniquement en mode N) 	SB 1221 200 kHz et SB 1223 200 kHz : Source
Tension nominale	24 V- à 4 mA, nominal	SB 24 V- : 24 V- à 7 mA, nominal SB 5 V- : 5 V- à 15 mA, nominal
Tension continue admise	30 V-, max.	SB 24 V- : 28,8 V- SB 5 V- : 6 V-
Tension de choc	35 V- pour 0,5 s	SB 24 V- : 35 V- pour 0,5 s SB 5 V- : 6 V
Signal 1 logique (min.)	15 V- à 2,5 mA	SB 24 V- : L+ moins 10 V- à 2,9 mA SB 5 V- : L+ moins 2,0 V- à 5,1 mA
Signal 0 logique (max.)	5 V- à 1 mA	SB 24 V- : L+ moins 5 V- à 1,4 mA SB 5 V- : L+ moins 1,0 V- à 2,2 mA
Isolation (site à logique)	500 V~ pendant 1 minute	500 V~ pendant 1 minute
Groupes d'isolation	<ul style="list-style-type: none"> CPU : 1 DI 8 SM 1221 : 2 DI 16 SM 1221 : 4 DI 2 SB 1223 : 1 SM 1223 : 2 	<ul style="list-style-type: none"> DI 4 SB 1221 : 1 DI 2 SB 1223 : 1
Temps de filtre	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 et 12,8 ms (sélectionnables par groupes de 4)	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 et 12,8 ms (sélectionnables par groupes de 4)

Caractéristiques techniques

A.4 Caractéristiques pour les entrées et sorties numériques

Caractéristiques techniques	CPU, SM et SB	SB rapide (200 kHz)
Nombre d'entrées simultanément	<ul style="list-style-type: none">• DI 8 SM 1221 et SM 1223 : 8• DI 16 SM 1221 et SM 1223 : 16• DI 2 SB 1223 : 2• CPU 1211C : 6 à 60 °C en position horizontale ou à 50 °C en position verticale• CPU 1212C : 4 (pas de points adjacents) à 60 °C en position horizontale ou à 50 °C en position verticale ; 8 à 55 °C en position horizontale ou à 45 °C en position verticale• CPU 1214C, CPU 1215C : 7 (pas de points adjacents) à 60 °C en position horizontale ou à 50 °C en position verticale ; 14 à 55 °C en position horizontale ou à 45 °C en position verticale• CPU 1217C : 5 entrées P/N (pas de points adjacents) et 4 entrées différentielles à 60 °C en position horizontale ou à 50 °C en position verticale ; 14 à 55 °C en position horizontale ou à 45 °C en position verticale	<ul style="list-style-type: none">• DI 4 SB 1221 : 4• DI 2 SB 1223 : 2
Longueur de câble (mètres)	<ul style="list-style-type: none">• 500 m blindé, 300 m non blindé• CPU : 50 m blindé pour HSC	50 m pair torsadée blindée

Remarque

Il est important dans le cas de fréquences de commutation supérieures à 20 kHz que les entrées numériques reçoivent une onde carrée. Tenez compte des options suivantes pour améliorer la qualité du signal arrivant aux entrées :

- Diminuez la longueur du câble.
- Remplacez une sortie exclusivement de type N par une sortie de type N et P
- Choisissez un câble de qualité supérieure
- Réduisez le circuit/les composants pour faire passer la tension de 24 V à 5 V (si le produit est conçu pour un fonctionnement à basse tension. Veuillez consulter les spécifications complètes dans le Manuel système de l'automate programmable S7-1200)
- Ajoutez une charge externe à l'entrée

Tableau A- 27 Fréquences d'entrée d'horloge HSC (max.)

Caractéristiques techniques	Monophasé	Quadrature de phase
CPU 1211C	100 kHz	80 kHz
CPU 1212C	100 kHz (la.0 à la.5) et 30 kHz (la.6 à la.7)	80 kHz (la.0 à la.5) et 20 kHz (la.6 à la.7)
CPU 1214C, CPU 1215C	100 kHz (la.0 à la.5) et 30 kHz (la.6 à lb.5)	80 kHz (la.0 à la.5) et 20 kHz (la.6 à lb.5)
CPU 1217C	1 MHz (lb.2 à lb.5) 100 kHz (la.0 à la.5) 30 kHz (la.6 à lb.1)	1 MHz (lb.2 à lb.5) 80 kHz (la.0 à la.5) 20 kHz (la.6 à lb.1)
SB rapides (200 kHz)	200 kHz	160 kHz
SB à vitesse standard	30 kHz	20 kHz

¹ Niveau 1 logique = 15 à 26 V-

A.4.2 Entrées TOR 120/230 V~

Tableau A- 28 Entrées TOR 120/230 V~

Caractéristiques techniques	SM
Type	CEI type 1
Tension nominale	120 V~ à 6 mA, 230 V~ à 9 mA
Tension continue admise	264 V~
Tension de choc	--
Signal 1 logique (min.)	79 V~ à 2,5 mA
Signal 0 logique (max.)	20 V~ à 1 mA
Courant de fuite (max.)	1 mA
Isolation (site à logique)	1500 V~ pour 1 minute
Groupes d'isolation ¹	4
Temps de retard d'entrée	<ul style="list-style-type: none"> Typique : 0,2 à 12,8 ms, personnalisable Maximum : --
Connexion de capteur de proximité à 2 fils (Bero) (max.)	1 mA
Longueur de câble	Non blindé
	Blindé
Nombre d'entrées simultanément à 1	8

¹ Les voies à l'intérieur d'un groupe doivent être de la même phase.

Caractéristiques techniques

A.4 Caractéristiques pour les entrées et sorties numériques

A.4.3 Sorties numériques (DQ)

Tableau A- 29Caractéristiques pour les sorties numériques (DQ)

Caractéristiques techniques	Relais (CPU et SM)	24 V CC (CPU, SM, et SB)	200 KHz 24 V CC (SB)
Type	Relais, contact sec	Transistor à effet de champ MOS (mode P)	Transistor à effet de champ MOS (mode N/P)
Plage de tension	5 à 30 V- ou 5 à 250 V~	20,4 à 28,8 V-	20,4 à 28,8 V- ¹ 4,25 à 6,0 V- ²
Signal 1 logique à courant max.	--	20 V- min.	L+ moins 1,5 V ¹ L+ moins 0,7 V ²
Signal 0 logique avec charge 10 kΩ	--	CPU : 20 V- min., 0,1 V- max. SB : 0,1 V- max. SM DC : 0,1 V- max.	1,0 V-, max. ¹ 0,2 V-, max. ²
Courant (max.)	2,0 A	0,5 A	0,1 A
Charge de lampe	30 W CC / 200 W CA	SB : 5 W	--
Résistance état activé	0,2 Ω max. lorsque neuf	0,6 Ω max.	11 Ω max. ¹ ou 7 Ω max. ²
Résistance état OFF	--	--	6 Ω max. ¹ ou 0,2 Ω max. ²
Courant de fuite par sortie	--	10 μA max.	--
Fréquence de sortie de trains d'impulsions	CPU : Sans objet ³	CPU : 100 kHz max., 2 Hz min. ⁴ SB : 20 kHz max., 2 Hz min. ⁵	200 kHz max., 2 Hz min.
Courant de choc	7 A avec contacts fermés	CPU : 8 A pour 100 ms max. SB : 5 A pour 100 ms max. SM : 8 A pour 100 ms max.	0,11 A
Protection contre la surcharge	Non	Non	Non
Isolation (site à logique)	Bobine à contact : 1500 V~ pour 1 minute Bobine à logique : Aucune	500 V~ pour 1 minute	500 V~ pour 1 minute
Groupes d'isolation	<ul style="list-style-type: none"> CPU 1211C : 1 CPU 1212C : 2 CPU 1214C : 2 CPU 1215C : 2 DQ 8 SM : 2 Commutation SM DQ 8 : 8 DQ 16 SM : 4 	<ul style="list-style-type: none"> CPU : 1 SB : 1 DQ 8 SM : 1 DQ 16 SM : 1 	1 ⁵
Résistance d'isolation	100 MΩ min. lorsque neuf	--	--
Isolation entre contacts ouverts	750 V~ pendant 1 minute	--	--
Nombre de sorties simultanément à 1	CPU 1211C : 4 à 60 °C en position horizontale ou à 50 °C en position verticale	--	--

Caractéristiques techniques	Relais (CPU et SM)	24 V CC (CPU, SM, et SB)	200 KHz 24 V CC (SB)
	CPU 1212C : 3 (pas de point adjacent) à 60 °C en position horizontale ou à 50 °C en position verticale ; 6 à 55 °C en position horizontale ou à 45 °C en position verticale	--	--
	CPU 1214C : 5 (pas de point adjacent) à 60 °C en position horizontale ou à 50 °C en position verticale ; 10 à 55 °C en position horizontale ou à 45 °C en position verticale	--	--
	CPU 1215C : 5 (pas de points adjacents) à 60 °C en position horizontale ou à 50 °C en position verticale ; 10 à 55 °C en position horizontale ou à 45 °C en position verticale	--	--
	CPU 1217C : 3 sorties état solide - MOSFET (source) (pas de point adjacent) et 4 sorties différentielles à 60 °C en position horizontale ou à 50 °C en position verticale 10 à 55 °C en position horizontale ou à 45 °C en position verticale	--	--
SM 1222 DQ8 : 8	Commutation SM1222 DQ 8 : 4 (pas de points adjacents) à 60 °C en position horizontale ou à 50 °C en position verticale, 8 à 55 °C en position horizontale ou à 45 °C en position verticale	SM 1222 DQ8 : 8	--
SM 1223 DI 8/DQ 8 relais : 8			
SM 1222 DQ16 : 8 (pas de points adjacents) à 60 °C en position horizontale ou à 50 °C en position verticale ; 16 à 55 °C en position horizontale ou à 45 °C en position verticale	SM 1222 DQ16 : 16	--	--
SM 1223 DI 16/DQ 16 relais : 8 (pas de points adjacents) à 60 °C en position horizontale ou à 50 °C en position verticale ; 16 à 55 °C en position horizontale ou à 45 °C en position verticale			
SM 1223 DI 8 x 120/230 VAC/DQ 8 relais : 4x SM 1223 DI 16/DQ 16 relais : 4 (pas de points adjacents) à 60 °C en position horizontale ou à 50 °C en position verticale ; 8 à 55 °C en position horizontale ou à 45 °C en position verticale	SM 1223 DI 8/DQ 8 : 8	SM 1223 DI 16/DQ 16 : 16	--

Caractéristiques techniques

A.4 Caractéristiques pour les entrées et sorties numériques

Caractéristiques techniques	Relais (CPU et SM)	24 V CC (CPU, SM, et SB)	200 KHz 24 V CC (SB)
		SB 1223 DI 1 DQ 2, SM 1223 DI2 DQ2 : 2	SB 1222 DQ 4 : 2 (pas de points adjacents) à 60 °C en position horizontale ou à 50 °C en position verticale ; 4 à 55 °C en position horizontale ou à 45 °C en position verticale
			SB 1222 DQ 4 x 5 VDC : 4
			SB 1223 D I 2/DQ 2 : 2
			SB 1223 DI 2/DQ 2 : 2
Courant par commun	Relais SM : <ul style="list-style-type: none"> SM 1222 DQ 8 et DQ 16 : 10 A Commutation SM 1222 DQ 8 : 2A SM 1223 DI 8/DQ 8:10 A SM 1223 DI 16/DQ 16 : 8 A SM 1223 DI 8x120/230 VAC/DQ 8 Rly : 10 	SM 24 VDC <ul style="list-style-type: none"> SM 1222 DQ 16: 8 A SM 1223 DI 8/DQ 8 : 4 A SM 1223 DI 16/DQ 16 : 8 A 	--
Tension de blocage inductive	--	L+ moins 48 V, dissipation 1 W	Aucune
Fréquence de commutation maximum des relais	1 Hz	--	--
Retard de commutation	10 ms max.	CPU : <ul style="list-style-type: none"> Qa.0 à Qa.3 : 1.0 µs max. de 0 à 1 ; 3.0 µs max. de 1 à 0 Qa.4 à Qb.1 : 50 µs max. de 0 à 1 ; 200 µs max. de 1 à 0 SB : 2 µs max. de 0 à 1 ; 10 µs max. de 1 à 0 SM : 50 µs max. de 0 à 1 ; 200 µs max. de 1 à 0 	1,5 µs + 300 s à la montée ¹ 1,5 µs + 300 ns à la retombée ¹ 200 µs + 300 ns à la montée ² 200 µs + 300 ns à la retombée ²
Durée d'utilisation mécanique (sans charge)	Relais : 10 000 000 cycles ouverture/fermeture	--	--
Durée d'utilisation des contacts à la charge nominale	Relais : 100 000 cycles ouverture/fermeture	--	--

Caractéristiques techniques	Relais (CPU et SM)	24 V CC (CPU, SM, et SB)	200 KHz 24 V CC (SB)
Comportement au passage de MARCHE à ARRET	Dernière valeur ou valeur de remplacement (valeur par défaut 0)	Dernière valeur ou valeur de remplacement (valeur par défaut 0)	Dernière valeur ou valeur de remplacement (valeur par défaut 0)
Longueur de câble (mètres)	500 m blindé, 150 m non blindé	500 m blindé, 150 m non blindé	50 m paire torsadée blindée

¹ SB 24 V- 200 kHz

² SB 5 V- 200 kHz

³ Pour les modèles de CPU avec sorties relais, vous devez installer un Signal Board (SB) numérique pour utiliser les sorties d'impulsions.

⁴ Selon votre receveur et câble d'impulsions, une résistance de charge supplémentaire (au moins 10% du courant nominal) peut améliorer la qualité du signal d'impulsion et l'immunité au bruit.

⁵ SB 1223 200 kHz DI 2 / DQ 2 : Pas d'isolation aux entrées

A.5 Modules E/S analogiques

Pour une liste plus complète des modules disponibles pour le S7-1200, reportez-vous au *Manuel Système de l'automate programmable S7-1200* ou au site web du service client (<http://www.siemens.com/tiaprofile>).

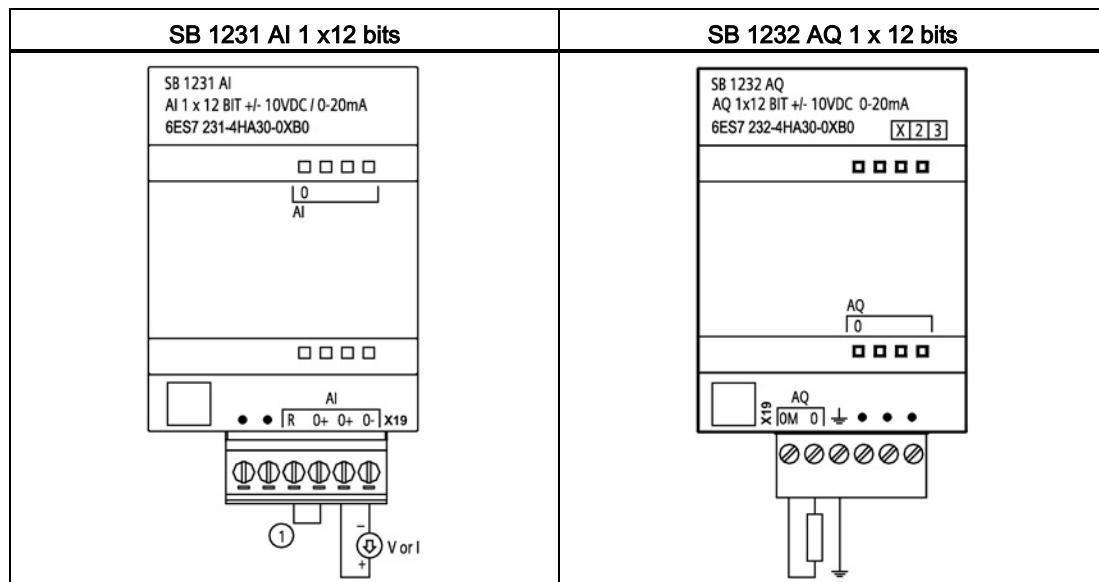
A.5.1 Entrée (AI) et sortie (AQ) analogiques SB 1231 et SB 1232

Tableau A- 30Caractéristiques générales

Caractéristiques techniques	SB 1231 AI 1 x12 bits ¹	SB 1232 AQ 1 x 12 bits
Référence	6ES7 231-4HA30-0XB0	6ES7 232-4HA30-0XB0
Dimensions L x H x P (mm)	38 x 62 x 21 mm	38 x 62 x 21 mm
Poids	35 grammes	40 grammes
Puissance dissipée	0,4 W	1,5 W
Consommation de courant (bus SM)	55 mA	15 mA
Consommation de courant (24 V-)	Aucune	40 mA (sans charge)
Nombre d'entrées-sorties	1	1
Type	Tension ou courant (différentiel)	Tension ou courant

¹ Le firmware de votre CPU doit être de V2.0 ou d'une version supérieure pour utiliser le SB 1231 AI 1 x analog input.

Tableau A- 31Schémas de câblage pour les SB analogiques



① Raccordez "R" et "0+" pour le courant

A.5.2 Entrée analogique (AI) SM 1231

Tableau A- 32 Entrées analogiques (AI) SM 1231

Caractéristiques techniques	SM 1231 AI 4 x 13 bits	AI 8 x 13 bit SM 1231	SM 1231 AI 4 x 16 bit
Numéro de référence (MLFB)	6ES7 231-4HD32-0XB0	6ES7 231-4HF32-0XB0	6ES7 231-5ND32-0XB0
Nombre d'entrées	4 entrées (AI)	8 entrées (AI)	4 entrées
Type	Tension ou courant (différentiel), sélectionnable par groupes de 2	Tension ou courant (différentiel), sélectionnable par groupes de 2	Tension ou courant (différentiel)
Dimensions L x H x P (mm)	45 x 100 x 75	45 x 100 x 75	45 x 100 x 75
Poids	180 grammes	180 grammes	180 grammes
Puissance dissipée	1,5 W	1,5 W	1,8 W
Consommation de courant (bus SM)	80 mA	90 mA	80 mA
Consommation de courant (24 V-)	45 mA	45 mA	65 mA

A.5.3 Sortie analogique (AQ) SM 1232

Tableau A- 33 Sorties analogiques (AQ) SM 1232

Caractéristiques techniques	SM 1232 AQ 2 x 14 bits	SM 1232 AQ 4 x 14 bits
Numéro de référence (MLFB)	6ES7 232-4HB32-0XB0	6ES7 232-4HD32-0XB0
Nombre et type de sorties	2 sorties (AQ)	4 sorties (AQ)
Dimensions L x H x P (mm)	45 x 100 x 75	45 x 100 x 75
Poids	180 grammes	180 grammes
Dissipation de courant	1,5 W	1,5 W
Consommation de courant (bus SM)	80 mA	80 mA
Consommation de courant (24 V-)	45 mA (sans charge)	45 mA (sans charge)

A.5.4 Entrée-sortie analogique SM 1234 (AI/AQ)

Tableau A- 34 Combinations entrée-sortie analogique SM 1234 (AI/AQ)

Caractéristiques techniques	AI 4 x 13 bit / AQ 2 x 14 bit SM 1234
Numéro de référence (MLFB)	6ES7 234-4HE32-0XB0
Nombre d'entrées	4 entrées (AI)
Type	Tension ou courant (différentiel), sélectionnable par groupes de 2
Nombre de sorties	2 sorties (AQ)
Type	Tension ou courant (différentiel)
Dimensions L x H x P (mm)	45 x 100 x 75
Poids	220 grammes

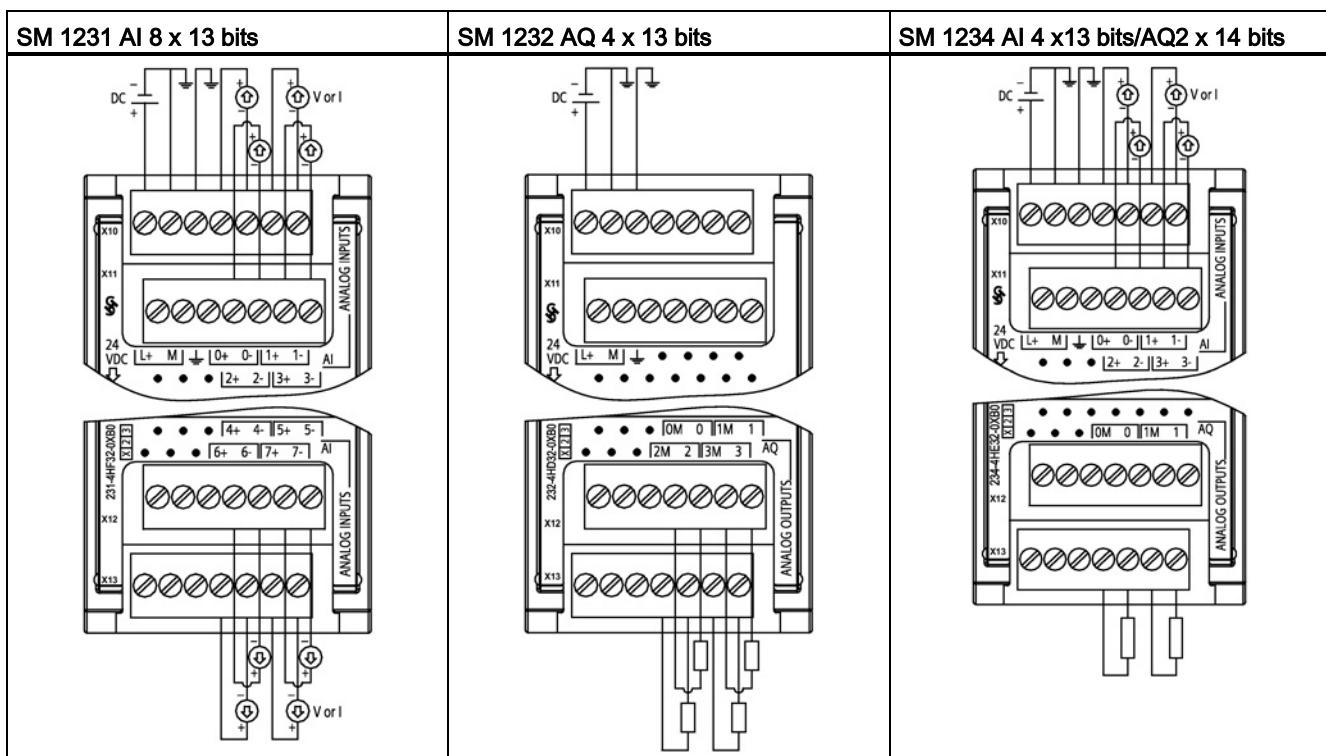
Caractéristiques techniques

A.5 Modules E/S analogiques

Caractéristiques techniques	AI 4 x 13 bit / AQ 2 x 14 bit SM 1234
Dissipation de courant	2,0 W
Consommation de courant (bus SM)	80 mA
Consommation de courant (24 V-)	60 mA (sans charge)

A.5.5 Schémas de câblage pour SM 1231 (AI), SM 1232 (AQ) et SM 1234 (AI/AQ)

Tableau A- 35Schémas de câblage pour les SM analogiques



Remarque

Les voies d'entrée de tension inutilisées doivent être court-circuitées.

Les voies d'entrée de courant inutilisées doivent être réglées sur la plage 0 à 20 mA et/ou la signalisation de l'erreur de rupture de fil doit être désactivée.

Les entrées configurées pour le mode courant ne conduiront pas à une régulation de courant, à moins que le module ne soit alimenté en électricité et configuré.

Les voies d'entrée de courant ne fonctionneront pas sauf si une alimentation électrique externe alimente le transmetteur.

A.6 battery board BB 1297

Battery Board BB 1297

Tableau A- 36Caractéristiques générales

Caractéristiques techniques	Battery Board BB 1297
Référence	6ES7 297-0AX30-0XA0
Dimensions L x H x P (mm)	38 x 62 x 21
Poids	28 grammes
Temps de conservation de l'horloge temps réel	Environ 1 an
Type de pile	CR1025 ¹
DEL "Maint" de la CPU	Signale qu'il faut remplacer la pile
Programme utilisateur	L'application / le système peut évaluer l'état de la pile

¹ Reportez-vous au *Manuel système de l'automate programmable S7-1200*, Chapitre 2, Installation pour obtenir des informations sur l'installation du BB 1297 ou le remplacement d'une batterie dans le BB.

On utilise le Battery Board BB 1297 pour les applications dans lesquelles la durée de conservation de l'horloge temps réel dépasse un mois. Le Battery Board BB 1297 présente les caractéristiques suivantes :

- Il prend en charge l'horloge temps réel lorsque l'API est hors tension. En combinaison avec le Battery Board BB 1297, la CPU S7-1200 assure la conservation de l'horloge temps réel pendant une période hors tension de l'application pouvant atteindre un an.
- Un seul Battery Board BB 1297 ou autre SB peut être utilisé à un moment donné.
- L'enfichage et le remplacement à chaud sont interdits. Le Battery Board BB 1297 ne peut être remplacé ou enfiché que lorsque la CPU est hors tension. Lorsqu'on retire le BB 1297 pour remplacer sa pile, alors que la CPU est hors tension, le supercondensateur interne conserve l'horloge temps réel pendant le remplacement de la pile.
- La DEL "Maint" de la CPU signale qu'il faut remplacer la pile.
- Le programme utilisateur permet de surveiller ou de contrôler l'état de la pile et un message utilisateur peut être affiché sur un appareil IHM ou le serveur Web.

A.7 Caractéristiques des E/S analogiques**A.7.1 Caractéristiques des entrées analogiques (CPU, SM et SB)**

Tableau A- 37Caractéristiques des entrées analogiques (AI)

Caractéristiques techniques	CPU	SB	SM
Type	Tension (mode simple)	Tension ou courant (différentiel)	Tension ou courant (différentiel), sélectionnable par groupes de 2
Plage	0 à 10 V	±10 V, ±5 V, ±2,5, 0 à 20 mA, ou 4 mA à 20 mA	±10 V, ±5 V, ±2,5 V, 0 à 20 mA, ou 4 mA à 20 mA
Résolution	10 bits	11 bits + bit de signe	12 bits + bit de signe
Plage de pleine échelle (mot de donnée)	0 à 27 648	-27 648 à 27 648	-27 648 à 27 648
Précision (25 °C / -20 à 60 °C)	3,0% / 3,5% de la pleine échelle	±0,3% / ±0,6% de la pleine échelle	±0,1% / ±0,2% de la pleine échelle
Plage de dépassement haut / bas (mot de donnée) (Voir note 1)	Tension : 27 649 à 32 511	Tension : 32 511 à 27 649 / -27 649 à -32 512	Tension : 32 511 à 27 649 / -27 649 à -32 512
	Courant : Sans objet	Courant : 32 511 à 27 649 / 0 à -4864	Courant : 32 511 à 27 649 / 0 à -4864
Débordement haut/bas (mot de donnée) (Voir note 1)	Tension : 32 512 à 32 767	Tension : 32 767 à 32 512 / -32 513 à -32 768	Tension : 32 767 à 32 512 / -32 513 à -32 768
	Courant : Sans objet	Courant : 32 767 à 32 512 / -4865 à -32 768	Courant : 32 767 à 32 512 / -4865 à -32 768
Tension/courant de résistance maximum	35 V- (tension)	±35 V / ±40 mA	±35 V / ±40 mA
Lissage (Voir note 2)	Aucun, faible, moyen ou fort	Aucun, faible, moyen ou fort	Aucun, faible, moyen ou fort
Antibruit (Voir note 2)	10, 50 ou 60 Hz	400, 60, 50 ou 10 Hz	400, 60, 50 ou 10 Hz
Principe de mesure	Conversion de valeur effective	Conversion de valeur effective	Conversion de valeur effective
Réjection en mode commun	Aucune	40 dB, CC pour 60 Hz	40 dB, CC pour 60 Hz
Plage de fonctionnement du signal (signal plus tension de mode commun)	Moins de +12 V et plus de 0 V	Moins de +35 V et plus de -35 V	Moins de +12 V et plus de -12 V

Caractéristiques techniques	CPU	SB	SM
Impédance de charge	à une extrémité : $\geq 100 \text{ k}\Omega$	Valeur différentielle : 220 $\text{k}\Omega$ (tension), 250 Ω (courant) Mode commun : 55 $\text{k}\Omega$ (tension), 55 Ω (courant)	Valeur différentielle : 9 $\text{M}\Omega$ (tension), 250 Ω (courant) Mode commun : 4,5 $\text{M}\Omega$ (tension), 4,5 Ω (courant)
Isolation (site à logique)	Aucune	Aucune	Aucune
Longueur de câble (mètres)	100 m, paire torsadée blindée	100 m, torsadé et blindé	100 m, torsadé et blindé
Diagnostic	Débordement haut/bas	Débordement haut/bas	Débordement haut/bas Basse tension 24 V-
Note 1 : Se reportez aux plages de mesure des entrées analogiques pour la tension et le courant (Page 417) pour déterminer les plages de débordement haut et bas et de dépassement haut et bas.			
Note 2 : Reportez-vous aux temps de réponse transitoire (Page 419) pour déterminer les valeurs de lissage et antibruit.			

A.7.2 Plages de mesures d'entrée (AI) pour tension et courant

Tableau A- 38 Représentation des entrées analogiques pour la tension (SB et SM)

Technologie		Plage de mesure de tension				
Décimal	Hexadécimal	$\pm 10 \text{ V}$	$\pm 5 \text{ V}$	$\pm 2,5 \text{ V}$	$\pm 1,25 \text{ V}$	
32767	7FFF ¹	11,851 V	5,926 V	2,963 V	1,481 V	Débordement haut
32512	7F00					
32511	7EFF	11,759 V	5,879 V	2,940 V	1,470 V	Plage de dépassement haut
27649	6C01					
27648	6C00	10 V	5 V	2,5 V	1,250 V	Plage nominale
20736	5100	7,5 V	3,75 V	1,875 V	0,938 V	
1	1	361,7 μV	180,8 μV	90,4 μV	45,2 μV	
0	0	0 V	0 V	0 V	0 V	
-1	FFFF					
-20736	AF00	-7,5 V	-3,75 V	-1,875 V	-0,938 V	
-27648	9400	-10 V	-5 V	-2,5 V	-1,250 V	
-27649	93FF					Plage de dépassement bas
-32512	8100	-11,759 V	-5,879 V	-2,940 V	-1,470 V	Débordement bas
-32513	80FF					
-32768	8000	-11,851 V	-5,926 V	-2,963 V	-1,481 V	

¹ 7FFF peut être renvoyé pour l'une des raisons suivantes : débordement haut (tel qu'indiqué dans ce tableau), avant que des valeurs valides soient disponibles (par exemple au moment d'une mise sous tension), ou si une rupture de fil est détectée.

Caractéristiques techniques

A.7 Caractéristiques des E/S analogiques

Tableau A- 39Représentation des entrées analogiques pour le courant (SB et SM)

Technologie		Plage de mesure de courant		
Décimal	Hexadécimal	0 mA à 20 mA	4 mA à 20 mA	
32767	7FFF	23,70 mA	22,96 mA	Débordement haut
32512	7F00			
32511	7EFF	23,52 mA	22,81 mA	Plage de dépassement haut
27649	6C01			
27648	6C00	20 mA	20 mA	Plage nominale
20736	5100	15 mA	16 mA	
1	1	723,4 nA	4 mA + 578,7 nA	
0	0	0 mA	4 mA	
-1	FFFF			Plage de dépassement bas
-4864	ED00	-3,52 mA	1,185 mA	
-4865	ECFF			Débordement bas
-32768	8000			

Tableau A- 40Représentation d'entrée analogique pour tension (CPU 1215C et CPU 1217C)

Système		Plage de mesure de tension		
Décimal	Hexadécimal	0 à 10 V		
32767	7FFF	11,851 V	Débordement haut	
32512	7F00			
32511	7EFF	11,759 V	Plage de dépassement haut	
27649	6C01			
27648	6C00	10 V	Plage nominale	
20736	5100	7,5 V		
34	22	12 mV		
0	0	0 V		
Valeurs négatives		Valeurs négatives impossibles		

A.7.3 Réponse transitoire pour les entrées analogiques (AI)

Le tableau suivant indique les temps de réponse transitoire pour les entrées analogiques (AI) de la CPU, du SB et du SM.

Tableau A- 41Réponse transitoire (ms) pour les entrées analogiques

Sélection avec lissage (moyennage d'échantillon)		Sélection de temps d'intégration			
		400 Hz (2.5 ms)	60 Hz (16.6 ms)	50 Hz (20 ms)	10 Hz (100 ms)
Aucun (1 cycle) : Pas de moyennage	CPU	Sans objet	63	65	130
	SB	4,5	18,7	22,0	102
	SM	4	18	22	100
Faible (4 cycles) : 4 échantillons	CPU	Sans objet	84	93	340
	SB	10,6	59,3	70,8	346
	SM	9	52	63	320
Moyen (16 cycles) : 16 échantillons	CPU	Sans objet	221	258	1210
	SB	33,0	208	250	1240
	SM	32	203	241	1200
Fort (32 cycles) : 32 échantillons	CPU	Sans objet	424	499	2410
	SB	63,0	408	490	2440
	SM	61	400	483	2410
Taux d'échantillon- nage	CPU	Sans objet	4,17	5	25
	SB	0,156	1,042	1,250	6,250

A.7.4 Temps d'échantillonnage et temps d'actualisation pour les entrées analogiques

Tableau A- 42Temps d'échantillonnage et temps d'actualisation pour SM et CPU

Fréquence de réjection (temps d'intégration)	Temps d'échantillon- nage	Temps d'actualisation pour toutes les voies		
		SM 4 voies	SM 8 voies	CPU AI
400 Hz (2.5 ms)	0,625 ms ¹	2,5 ms	10 ms	Non disponible ms
60 Hz (16.6 ms)	4,170 ms	4,17 ms	4,17 ms	4,17 ms
50 Hz (20 ms)	5,000 ms	5 ms	5 ms	5 ms
10 Hz (100 ms)	25,000 ms	25 ms	25 ms	25 ms

¹ La fréquence d'échantillonnage pour un SM 8 voies est de 1,250 ms.

Tableau A- 43Temps d'échantillonnage et temps d'actualisation pour SB

Fréquence de réjection (temps d'intégration)	Temps d'échantillonnage	Temps d'actualisation du SB
400 Hz (2.5 ms)	0,156 ms	0,156 ms
60 Hz (16.6 ms)	1,042 ms	1,042 ms

Caractéristiques techniques

A.7 Caractéristiques des E/S analogiques

Fréquence de réjection (temps d'intégration)	Temps d'échantillonnage	Temps d'actualisation du SB
50 Hz (20 ms)	1,250 ms	1,25 ms
10 Hz (100 ms)	6,250 ms	6,25 ms

A.7.5 Caractéristiques des sorties analogiques

Tableau A- 44Caractéristiques des sorties analogiques (SM et SB)

Caractéristiques techniques	SB	SM
Type	Tension ou courant	Tension ou courant
Plage	± 10 V, 0 à 20 mA ou 4 à 20 mA	± 10 V, 0 à 20 mA ou 4 à 20 mA
Résolution	Tension : 12 bits Courant : 11 bits	Tension : 14 bits Courant : 13 bits
Plage de pleine échelle (mot de donnée) (Voir note 1)	Tension : -27 648 à 27 648 Courant : 0 à 27 648	Tension : -27 648 à 27 648 Courant : 0 à 27 648
Précision (25 °C / -20 à 60 °C)	$\pm 0,5\%$ / $\pm 1\%$ de la pleine échelle	$\pm 0,3\%$ / $\pm 0,6\%$ de la pleine échelle
Durée d'établissement (95 % de la nouvelle valeur)	Tension : 300 μ S (R), 750 μ S (1 μ F) Courant : 600 μ s (1 mH), 2 ms (10 mH)	Tension : 300 μ S (R), 750 μ S (1 μ F) Courant : 600 μ s (1 mH), 2 ms (10 mH)
Impédance de charge	Tension : $\geq 1\,000\,\Omega$ Courant : $\leq 600\,\Omega$	Tension : $\geq 1\,000\,\Omega$ Courant : $\leq 600\,\Omega$
Comportement au passage de MARCHE à ARRET	Dernière valeur ou valeur de remplacement (valeur par défaut 0)	Dernière valeur ou valeur de remplacement (valeur par défaut 0)
Isolation (site à logique)	Aucune	Aucune
Longueur de câble (mètres)	100 m, torsadé et blindé	100 m, torsadé et blindé
Diagnostic	<ul style="list-style-type: none"> Débordement haut/bas Court-circuit à la terre (mode tension uniquement) Rupture de fil (mode courant uniquement) 	<ul style="list-style-type: none"> Débordement haut/bas Court-circuit à la terre (mode tension uniquement) Rupture de fil (mode courant uniquement) Basse tension 24 V-

Note 1 : Reportez-vous aux plages de sortie pour la tension et le courant (Page 421) pour la plage de pleine échelle.

A.7.6 Plages de mesures de sortie (AQ) pour tension et courant

Tableau A- 45Représentation de sorties analogiques pour tension (SB et SM)

Système		Plage de sortie de tension	
Décimal	Hexadécimal	$\pm 10 \text{ V}$	
32767	7FFF	Voir note 1	Débordement haut
32512	7F00	Voir note 1	
32511	7EFF	11,76 V	Plage de dépassement haut
27649	6C01		
27648	6C00	10 V	Plage nominale
20736	5100	7,5 V	
1	1	361,7 μV	
0	0	0 V	
-1	FFFF	-361,7 μV	
-20736	AF00	-7,5 V	
-27648	9400	-10 V	
-27649	93FF		Plage de dépassement bas
-32512	8100	-11,76 V	
-32513	80FF	Voir note 1	Débordement bas
-32768	8000	Voir note 1	

¹ À l'état débordement haut ou bas, les sorties analogiques prendront la valeur de remplacement de l'état de fonctionnement Arrêt.

Tableau A- 46Représentation de sorties analogiques pour courant (SB et SM)

Système		Plage de sortie du courant		
Décimal	Hexadécimal	0 mA à 20 mA	4 mA à 20 mA	
32767	7FFF	Voir note 1	Voir note 1	Débordement haut
32512	7F00	Voir note 1	Voir note 1	
32511	7EFF	23,52 mA	22,81 mA	Plage de dépassement haut
27649	6C01			
27648	6C00	20 mA	20 mA	Plage nominale
20736	5100	15 mA	16 mA	
1	1	723,4 nA	4 mA + 578,7 nA	
0	0	0 mA	4mA	
-1	FFFF		4 mA à 578,7 nA	Plage de dépassement bas
-6912	E500		0 mA	

Caractéristiques techniques

A.7 Caractéristiques des E/S analogiques

Système		Plage de sortie du courant		
Décimal	Hexadécimal	0 mA à 20 mA	4 mA à 20 mA	
-6913	E4FF			Impossible. Valeur de sortie limitée à 0 mA.
-32512	8100			
-32513	80FF	Voir note 1	Voir note 1	Débordement bas
-32768	8000	Voir note 1	Voir note 1	

¹ À l'état débordement haut ou bas, les sorties analogiques prendront la valeur de remplacement de l'état de fonctionnement Arrêt.

Tableau A- 47 Représentation de sorties analogiques pour courant (CPU 1215C et CPU 1217C)

Système		Plage de sortie du courant	
Décimal	Hexadécimal	0 mA à 20 mA	
32767	7FFF	Voir note 1	Débordement haut
32512	7F00	Voir note 1	
32511	7EFF	23,52 mA	Plage de dépassement haut
27649	6C01		
27648	6C00	20 mA	Plage nominale
20736	5100	15 mA	
34	22	0,0247 mA	
0	0	0 mA	
Valeurs négatives		Valeurs négatives impossibles	

¹ En cas de débordement haut, les sorties analogiques se comportent conformément aux paramétrages des propriétés de configuration d'appareil. Pour le paramètre "Réaction à l'arrêt de la CPU", sélectionnez : "Appliquer valeur de remplacement" ou "Conserver dernière valeur".

A.8 Modules RTD et Thermocouple

Les modules thermocouple (TC) (TC SB 1231 et TC SM 1231) mesurent la valeur de la tension connectée aux entrées analogiques. Cette valeur peut correspondre soit à une température provenant d'un TC, soit à des volts.

- S'il s'agit d'une tension, la valeur de la plage pleine échelle nominale sera 27 648 en décimal.
- S'il s'agit d'une température, la valeur sera notifiée en degrés multipliés par dix (par exemple, 25,3 degrés seront indiqués sous la forme 253 en décimal).

Les modules RTD (RTD SB 1231 et RTD SM 1231) mesurent la valeur de la résistance connectée aux entrées analogiques. Cette valeur peut correspondre soit à une température, soit à une résistance.

- S'il s'agit d'une résistance, la valeur pleine échelle de la plage nominale sera 27 648 en décimal.
- S'il s'agit d'une température, la valeur sera notifiée en degrés multipliés par dix (par exemple, 25,3 degrés seront indiqués sous la forme 253 en décimal).

Les modules RTD prennent en charge les mesures avec des connexions à 2, 3 et 4 câbles à la résistance de capteur.

Remarque

Les modules RTD et TC reportent 32 767 sur un canal activé sans capteur connecté. Si la détection de câble ouvert est également activée, le module fait clignoter les DEL appropriées.

La meilleure précision est obtenue pour les plages RTD $10\ \Omega$ avec des montages 4 fils.

La résistance des câbles de connexion en mode bifilaire provoquera une erreur dans la lecture du capteur et par conséquent la précision n'est pas garantie.

Remarque

Une fois le courant appliqué, le module réalise une calibration interne pour le convertisseur analogique-numérique. Pendant ce temps, il émet une valeur de 32 767 sur chaque voie jusqu'à ce que des données valables soient disponibles sur cette voie. Votre programme utilisateur peut nécessiter de permettre ce temps d'initialisation. Comme la configuration du module peut faire varier la longueur du temps d'initialisation, vous devez vérifier le comportement du module dans votre configuration. Si nécessaire, vous pouvez inclure une logique dans votre programme utilisateur pour contenir le temps d'initialisation du module.

A.8.1 Caractéristiques RTD SB 1231 et TC SB 1231**Remarque**

Le firmware de votre CPU doit être de version V2.0 ou plus pour que vous puissiez utiliser ces SB TC et RTD.

Tableau A- 48Caractéristiques générales

Caractéristiques techniques	SB 1231 AI 1 x16 bits TC	SB 1231 AI 1 x 16 bits RTD
Référence	6ES7 231-5QA30-0XB0	6ES7 231-5PA30-0XB0
Dimensions L x H x P (mm)	38 x 62 x 21 mm	38 x 62 x 21 mm
Poids	35 grammes	35 grammes
Dissipation de courant	0,5 W	0,7 W
Consommation de courant (bus SM)	5 mA	5 mA
Consommation de courant (24 V-)	20 mA	25 mA
Nombre d'entrées (Page 428)	1	1
Type	TC flottant et mV	RTD et Ω référencés par module
Diagnostic	<ul style="list-style-type: none">• Dépassement haut / bas^{1, 2}• Rupture de fil³	<ul style="list-style-type: none">• Dépassement haut / bas^{1, 2}• Rupture de fil³

¹ Les informations d'alarme de diagnostic de dépassement haut et bas seront rapportées aux valeurs de données analogiques si les alarmes sont désactivées dans la configuration du module.

² RTD : Pour les plages de résistance, la détection de débordement bas n'est jamais activée.

³ Lorsque l'alarme de rupture de fil est désactivée et qu'un fil est ouvert dans le câblage du capteur, le module peut émettre des valeurs aléatoires.

Tableau A- 49 Schémas de câblage pour TC et RTD SB 1231

SB 1231 AI 1 x 16 bits TC	SB 1231 AI 1 x 16 bits RTD
<p>SB 1231 TC AI 1 x TC x 16 BIT 6ES7 231-5QA30-0XB0</p> <p>SB 1231 RTD AI 1 x RTD x 16 BIT 6ES7 231-5PA30-0XB0</p> <p>AI</p> <p>AI</p> <p>AI0 - TC</p> <p>AI0 - RTD</p> <p>AI 19</p> <p>M+ M- I+ I-</p> <p>② RTD 2 fils ③ RTD 3 fils ④ RTD 4 fils</p>	

① Bouclage sur l'entrée RTD inutilisée

② RTD 2 fils ③ RTD 3 fils ④ RTD 4 fils

A.8.2 Caractéristiques RTD SM 1231

Tableau A- 50 Caractéristiques générales

Caractéristiques techniques	AI 4 x RTD x 16 bit SM 1231	SM 1231 AI 8 x RTD x 16 bits
Référence	6ES7 231-5PD32-0XB0	6ES7 231-5PF32-0XB0
Dimensions L x H x P (mm)	45 x 100 x 75	70 x 100 x 75
Poids	220 grammes	270 grammes
Dissipation de courant	1,5 W	1,5 W
Consommation de courant (bus SM)	80 mA	90 mA
Consommation de courant ¹ (24 V-)	40 mA	40 mA

Caractéristiques techniques

A.8 Modules RTD et Thermocouple

Caractéristiques techniques	AI 4 x RTD x 16 bit SM 1231	SM 1231 AI 8 x RTD x 16 bits
Nombre d'entrées (Page 428)	4	8
Type	RTD et Ω référencés par module	RTD et Ω référencés par module
Diagnostic	<ul style="list-style-type: none"> Débordement haut/bas^{2,3} 24 V- basse tension² Rupture de fil (mode courant uniquement)⁴ 	<ul style="list-style-type: none"> Débordement haut/bas^{2,3} 24 V- basse tension² Rupture de fil (mode courant uniquement)⁴

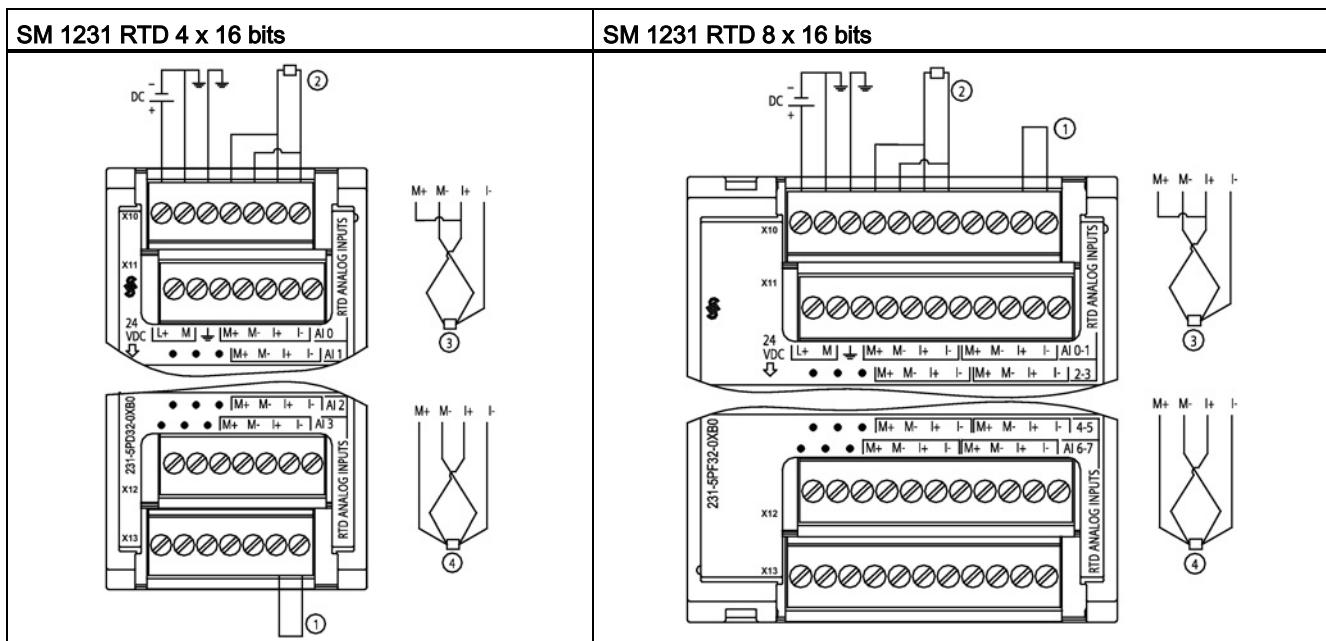
¹ 20,4 à 28,8 V- (classe 2, puissance limitée ou alimentation de capteur de la CPU)

² Les informations d'alarmes de diagnostic débordement haut, débordement bas et basse tension sont notifiées dans les valeurs de données analogiques même si les alarmes sont désactivées dans la configuration du module.

³ La détection de débordement bas n'est jamais activée pour les plages de résistance.

⁴ Lorsque l'alarme de rupture de fil est désactivée et qu'un fil est ouvert dans le câblage du capteur, le module peut émettre des valeurs aléatoires.

Tableau A- 51Schémas de câblage pour les SM RTD



(1) Bouclage sur les entrées RTD inutilisées

(2) RTD 2 fils

(3) RTD 3 fils

(4) RTD 4 fils

Note : les connecteurs doivent être en or. Voir le *Manuel système de l'automate programmable S7-1200*, annexe C.

A.8.3 Caractéristiques TC SM 1231

Tableau A- 52Caractéristiques générales

Modèle	SM 1231 AI 4 x 16 bits TC	SM 1231 AI 8 x 16 bits TC
Référence	6ES7 231-5QD32-0XB0	6ES7 231-5QF32-0XB0
Dimensions L x H x P (mm)	45 x 100 x 75	45 x 100 x 75
Poids	180 grammes	Xxx grammes
Dissipation de courant	1,5 W	1,5 W
Consommation de courant (bus SM)	80 mA	80 mA
Consommation de courant ¹ (24 V-)	40 mA	40 mA
Nombre d'entrées (Page 428)	4	8
Type	TC flottant et mV	TC flottant et mV
Diagnostic	<ul style="list-style-type: none"> • Débordement haut/bas² • 24 V- basse tension² • Rupture de fil (mode courant uniquement)³ 	<ul style="list-style-type: none"> • Débordement haut/bas² • 24 V- basse tension² • Rupture de fil (mode courant uniquement)³

¹ 20,4 à 28,8 V- (classe 2, puissance limitée ou alimentation de capteur de la CPU)

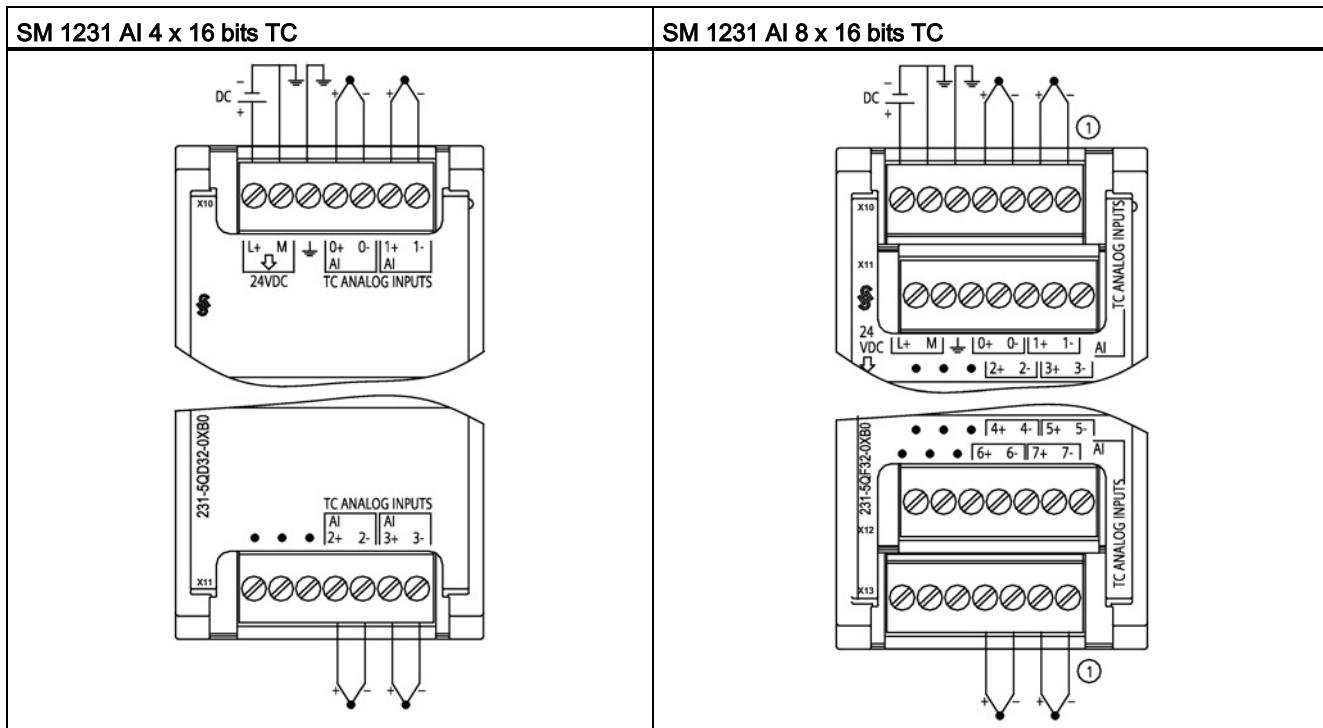
² Les informations d'alarmes de diagnostic débordement haut, débordement bas et basse tension sont notifiées dans les valeurs de données analogiques même si les alarmes sont désactivées dans la configuration du module.

³ Lorsque l'alarme de rupture de fil est désactivée et qu'un fil est ouvert dans le câblage du capteur, le module peut émettre des valeurs aléatoires.

Caractéristiques techniques

A.8 Modules RTD et Thermocouple

Tableau A- 53Schémas de câblage pour les SM TC



(1) SM 1231 AI 8 TC : Pour plus de clarté, les TC 2, 3, 4 et 5 ne sont pas connectés sur l'illustration.

A.8.4 Caractéristiques d'entrée analogique pour RTD et TC (SM et SB)

Tableau A- 54Entrées analogiques pour les modules RTD et TC (SM et SB)

Caractéristiques techniques		RTD et Thermocouple (TC)
Nombre d'entrées		1 (SB), 4 ou 8 (SM)
Type		<ul style="list-style-type: none"> RTD : RTD et Ω référencés par module TC : TC flottant et mV
Plage		Voir les tables de type RTD/TC : <ul style="list-style-type: none"> RTD (Page 431) TC (Page 429)
Résolution	Température	0,1 °C / 0,1 °F
	Résistance / tension	15 bits plus signe
Tension de résistance maximum		± 35 V
Antibruit		85 dB pour le réglage de filtre sélectionné (10 Hz, 50 Hz, 60 Hz ou 400 Hz)
Réjection en mode commun		> 120 dB à 120 V~
Impédance		≥ 10 M Ω

Caractéristiques techniques		RTD et Thermocouple (TC)
Isolation	Site à logique	500 V~
	Site à 24 V-	SM RTD et SM TC : 500 V~ (non applicable pour les SB RTD et SB TC)
	24 V- à logique	SM RTD et SM TC : 500 V~ (non applicable pour les SB RTD et SB TC)
Isolation voie à voie		<ul style="list-style-type: none"> SM RTD : Aucune (non applicable pour SB RTD) SM TC : 120 V~ (non applicable pour SB TC)
Précision (25 °C / -20 à 60 °C)		Voir les tables de type RTD/TC : <ul style="list-style-type: none"> RTD (Page 431) TC (Page 429)
Répétabilité		± 0,05% pleine échelle
Dissipation maximale du capteur		<ul style="list-style-type: none"> RTD : 0,5 mW TC : Sans objet
Principe de mesure		Intégration
Temps d'actualisation du module		Voir les tables de sélection de filtre RTD/TC : <ul style="list-style-type: none"> RTD (Page 432) TC (Page 430)
Erreur de soudure froide		<ul style="list-style-type: none"> RTD : Sans objet TC : ±1,5 °C
Longueur de câble (mètres)		max. 100 mètres au capteur
Résistance de fil		<ul style="list-style-type: none"> RTD : 20 Ω, 2,7 Ω pour RTD 10 Ω max. TC : 100 Ω max.

A.8.5 Type de thermocouple

Tableau A- 55Types de thermocouple (plages et précision)

Type	Dépassement bas minimum ¹	Limite basse plage nominale	Limite basse plage nominale	Dépassement haut maximum ²	Précision en plage normale ^{3, 4} à 25 °C	Précision en plage normale ^{3, 4} -20 °C à 60 °C
J	-210,0 °C	-150,0 °C	1200,0 °C	1450,0 °C	±0,3 °C	±0,6 °C
K	-270,0 °C	-200,0 °C	1372,0 °C	1622,0 °C	±0,4 °C	±1,0 °C
T	-270,0 °C	-200,0 °C	400,0 °C	540,0 °C	±0,5 °C	±1,0 °C
E	-270,0 °C	-200,0 °C	1000,0 °C	1200,0 °C	±0,3 °C	±0,6 °C
R & S	-50,0 °C	100,0 °C	1768,0 °C	2019,0 °C	±1,0 °C	±2,5 °C
B	0,0 °C	200,0 °C	800,0 °C	--	±2,0 °C	±2,5 °C
	--	800,0 °C	1820,0 °C	1820,0 °C	±1,0 °C	±2,3 °C
N	-270,0 °C	-200,0 °C	1300,0 °C	1550,0 °C	±1,0 °C	±1,6 °C

Caractéristiques techniques

A.8 Modules RTD et Thermocouple

Type	Dépassement bas minimum ¹	Limite basse plage nominale	Limite basse plage nominale	Dépassement haut maximum ²	Précision en plage normale ^{3, 4} à 25 °C	Précision en plage normale ^{3, 4} -20 °C à 60 °C
C	0,0 °C	100,0 °C	2315,0 °C	2500,0 °C	±0,7 °C	±2,7 °C
TXK/XK(L)	-200,0 °C	-150,0 °C	800,0 °C	1050,0 °C	±0,6 °C	±1,2 °C
Tension	-32512	-27648 -80 mV	27648	32511	±0,05 %	±0,1 %

¹ Les valeurs de thermocouple inférieures à la valeur de dépassement bas minimum sont signalées en tant que -32768.

² Les valeurs de thermocouple supérieures à la valeur de dépassement haut maximum sont signalées comme 32767.

³ L'erreur de soudure froide interne est de ±1,5 °C pour toutes les plages. Cette valeur s'ajoute à l'erreur indiquée dans ce tableau. Le module requiert au minimum 30 minutes d'échauffement pour atteindre cette spécification.

⁴ Uniquement pour SM TC 4 voies : La précision peut être réduite en présence de radiofréquence rayonnée entre 970 MHz et 990 MHz.

Remarque

Canal de thermocouple

Chaque canal sur le module d'entrées - sorties Thermocouple peut être configuré avec un type de thermocouple différent (sélectionnable dans le logiciel pendant la configuration du module).

A.8.6 Sélection de filtres de thermocouple et de temps d'actualisation

Pour les thermocouples mesurant, il est recommandé d'utiliser un temps d'intégration de 100 ms. Le choix de temps d'intégration inférieurs augmentera l'erreur de répétabilité des températures lues.

Tableau A- 56 Sélection de filtres de thermocouple et de temps d'actualisation

Fréquence de réjection (Hz)	Temps d'intégration (ms)	Temps de mise à jour (secondes)		
		SB une voie	SM 4 voies	SM 8 voies
10	100	0.301	1.225	2.450
50	20	0.061	0.263	0.525
60	16.67	0.051	0.223	0.445
400 ¹	10	0.031	0.143	0.285

¹ Le temps d'intégration est de 10 ms pour conserver la résolution et la précision du module lorsque la réjection 400 Hz est sélectionnée. Cette sélection correspond également à une réjection des bruits 100 Hz et 200 Hz.

A.8.7 Table de sélection type de capteur RTD

Tableau A- 57 Plages et précision pour les différents capteurs supportés par les modules RTD

Coefficient de température	Type de RTD	Dépassement bas minimum ¹	Limite basse plage nominale	Limite haute plage nominale	Dépassement haut maximum ²	Précision de plage normale à 25 °C	Précision de plage normale de -20 °C à 60 °C
Pt 0.003850 ITS90 DIN EN 60751	Pt 100 climatique	-145,00 °C	-120,00 °C	145,00 °C	155,00 °C	±0,20 °C	±0,40 °C
	Pt 10	-243,0 °C	-200,0 °C	850,0 °C	1000,0 °C	±1,0 °C	±2,0 °C
	Pt 50	-243,0 °C	-200,0 °C	850,0 °C	1000,0 °C	±0,5 °C	±1,0 °C
	Pt 100						
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0.003902 Pt 0.003916 Pt 0.003920	Pt 100	-243,0 °C	-200,0 °C	850,0 °C	1000,0 °C	±0,5 °C	±1,0 °C
	Pt 200	-243,0 °C	-200,0 °C	850,0 °C	1000,0 °C	±0,5 °C	±1,0 °C
	Pt 500						
	Pt 1000						
Pt 0.003910	Pt 10	-273,2 °C	-240,0 °C	1100,0 °C	1295 °C	±1,0 °C	±2,0 °C
	Pt 50	-273,2 °C	-240,0 °C	1100,0 °C	1295 °C	±0,8 °C	±1,6 °C
	Pt 100						
	Pt 500						
Ni 0.006720 Ni 0.006180	Ni 100	-105,0 °C	-60,0 °C	250,0 °C	295,0 °C	±0,5 °C	±1,0 °C
	Ni 120						
	Ni 200						
	Ni 500						
	Ni 1000						
LG-Ni 0.005000	LG-Ni 1000	-105,0 °C	-60,0 °C	250,0 °C	295,0 °C	±0,5 °C	±1,0 °C
Ni 0.006170	Ni 100	-105,0 °C	-60,0 °C	180,0 °C	212,4 °C	±0,5 °C	±1,0 °C
Cu 0.004270	Cu 10	-240,0 °C	-200,0 °C	260,0 °C	312,0 °C	±1,0 °C	±2,0 °C
Cu 0.004260	Cu 10	-60,0 °C	-50,0 °C	200,0 °C	240,0 °C	±1,0 °C	±2,0 °C
	Cu 50						
	Cu 100						
Cu 0.004280	Cu 10	-240,0 °C	-200,0 °C	200,0 °C	240,0 °C	±1,0 °C	±2,0 °C
	Cu 50	-240,0 °C	-200,0 °C	200,0 °C	240,0 °C	±0,7 °C	±1,4 °C
	Cu 100						

¹ Les valeurs RTD inférieures à la valeur de dépassement bas minimum sont signalées en tant que -32768.² Les valeurs RTD supérieures à la valeur de dépassement haut maximum sont signalées en tant que +32767.

Caractéristiques techniques

A.8 Modules RTD et Thermocouple

Tableau A- 58Résistance

Plage	Dépassement bas minimum	Limite basse plage nominale	Limite haute plage nominale	Dépassement haut maximum ¹	Précision de plage normale à 25 °C	Précision de plage normale de -20 °C à 60 °C
150 Ω	Sans objet	0 (0 Ω)	27648 (150 Ω)	176,383 Ω	±0,05 %	±0,1 %
300 Ω	Sans objet	0 (0 Ω)	27648 (300 Ω)	352,767 Ω	±0,05 %	±0,1 %
600 Ω	Sans objet	0 (0 Ω)	27648 (600 Ω)	705,534 Ω	±0,05 %	±0,1 %

¹ Les valeurs de résistance supérieures à la valeur de dépassement haut maximum sont signalées en tant que 32767.

A.8.8 Sélection de filtres de thermocouple et de temps d'actualisation

Tableau A- 59Sélection de filtres de thermocouple et de temps d'actualisation

Fréquence de réjection des bruits (Hz)	Temps d'intégration (ms)	Temps d'actualisation (secondes)		
		SB une voie	SM 4 voies	SM 8 voies
10	100	4 / 2 fils : 0,301	4 / 2 fils : 1,222	4 / 2 fils : 2,445
		3 fils : 0,601	3 fils : 2,445	3 fils : 4,845
50	20	4 / 2 fils : 0,061	4 / 2 fils : 0,262	4 / 2 fils : 0,525
		3 fils : 0,121	3 fils : 0,505	3 fils : 1,015
60	16,67	4 / 2 fils : 0,051	4 / 2 fils : 0,222	4 / 2 fils : 0,445
		3 fils : 0,101	3 fils : 0,424	3 fils : 0,845
400 ¹	10	4 / 2 fils : 0,031	4 / 2 fils : 0,142	4 / 2 fils : 0,285
		3 fils : 0,061	3 fils : 0,264	3 fils : 0,525

¹ Le temps d'intégration est de 10 ms pour conserver la résolution et la précision du module lorsque le filtre 400 Hz est sélectionné. Cette sélection correspond également à une réjection des bruits 100 Hz et 200 Hz.

Remarque

Le module signale 32767 sur toute voie activée à laquelle aucun capteur n'est connecté. Si la détection de câble ouvert est également activée, le module fait clignoter les DEL rouges appropriées.

La meilleure précision est obtenue pour les plages RTD 10 Ω avec des montages 4 fils.

La résistance des fils de connexion en montage 2 fils provoquera une erreur dans la lecture du capteur. La précision ne peut donc pas être garantie.

A.9 Interfaces de communication

Pour une liste plus complète des modules disponibles pour le S7-1200, reportez-vous au *Manuel Système de l'automate programmable S7-1200* ou au site web du service client (<http://www.siemens.com/tiportal>).

A.9.1 Maître/esclave PROFIBUS

A.9.1.1 ESCLAVE PROFIBUS DP CM 1242-5

Tableau A- 60Caractéristiques techniques du CM 1242-5

Caractéristiques techniques	
Référence	6GK7 242-5DX30-0XE0
Interfaces	
Connexion à PROFIBUS	Connecteur femelle subD 9 broches
Consommation maximale de courant sur l'interface PROFIBUS lors de la connexion de composantes réseau (par exemple composantes de réseau optique)	15 mA à 5 V (uniquement pour terminaison de bus) *)
Conditions ambiantes autorisées	
Température ambiante <ul style="list-style-type: none"> pendant le stockage pendant le transport pendant le fonctionnement avec une installation verticale (profilé support horizontal) pendant le fonctionnement avec une installation horizontale (profilé support vertical) 	<ul style="list-style-type: none"> -40 °C à 70 °C -40 °C à 70 °C 0 °C à 55 °C 0 °C à 45 °C
Humidité relative à 25 °C durant le fonctionnement, sans condensation, maximum	95 %
Degré de protection	IP20
Alimentation, consommation de courant et dissipation de courant	
Type d'alimentation	Courant continu
Alimentation électrique depuis le bus interne	5 V
Consommation de courant (typique)	150 mA
Perte de puissance effective (typique)	0,75 W
Séparation galvanique <ul style="list-style-type: none"> Interface PROFIBUS à la terre Interface PROFIBUS au circuit interne 	710 V CC pendant une minute
Dimensions et poids	
<ul style="list-style-type: none"> Largeur Hauteur Profondeur 	<ul style="list-style-type: none"> 30 mm 100 mm 75 mm

Caractéristiques techniques

A.9 Interfaces de communication

Caractéristiques techniques	
Poids	
• Poids net • Poids avec l'emballage	• 115 g • 152 g

*La charge actuelle d'un client externe connecté entre VP (broche 6) et DGND (broche 5) ne doit pas dépasser un maximum de 15 mA (résistant au court-circuit) pour la terminaison de bus.

A.9.1.2 Brochage de la prise subD du CM 1242-5

Interface PROFIBUS

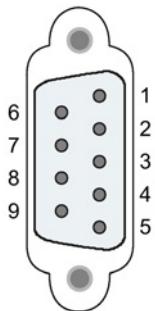


Tableau A- 61Brochage de la prise subD

Broche	Description	Broche	Description
1	- non utilisée -	6	P5V2 : alimentation électrique +5V
2	- non utilisée -	7	- non utilisée -
3	RxD/TxD-P : Ligne de données B	8	RxD/TxD-N : Ligne de données A
4	RTS	9	- non utilisée -
5	M5V2 : Potentiel de référence de données (DGND terre)	Boîtier	Raccord de mise à la terre

A.9.1.3 Maître PROFIBUS DP CM 1243-5

Tableau A- 62Caractéristiques techniques du CM 1243-5

Caractéristiques techniques	
Référence	6GK7 243-5DX30-0XE0
Interfaces	
Connexion à PROFIBUS	Connecteur femelle subD 9 broches
Consommation maximale de courant sur l'interface PROFIBUS lors de la connexion de composantes réseau (par exemple composantes de réseau optique)	15 mA à 5 V (uniquement pour terminaison de bus) *)
Conditions ambiantes autorisées	
Température ambiante <ul style="list-style-type: none"> pendant le stockage pendant le transport pendant le fonctionnement avec une installation verticale (profilé support horizontal) pendant le fonctionnement avec une installation horizontale (profilé support vertical) 	<ul style="list-style-type: none"> -40 °C à 70 °C -40 °C à 70 °C 0 °C à 55 °C 0 °C à 45 °C
Humidité relative à 25 °C durant le fonctionnement, sans condensation, maximum	95 %
Degré de protection	IP20
Alimentation, consommation de courant et dissipation de courant	
Type d'alimentation	Courant continu
Alimentation électrique / externe <ul style="list-style-type: none"> minimum maximum 	24 V <ul style="list-style-type: none"> 19,2 V 28,8 V
Consommation de courant (typique) <ul style="list-style-type: none"> à partir de 24 V CC depuis le bus interne du S7-1200 	<ul style="list-style-type: none"> 100 mA 0 mA
Perte de puissance effective (typique) <ul style="list-style-type: none"> à partir de 24 V CC depuis le bus interne du S7-1200 	<ul style="list-style-type: none"> 2,4 W 0 W
Alimentation 24 V CC / externe <ul style="list-style-type: none"> Section du câble min. Section du câble max. Couple de serrage des bornes à vis 	<ul style="list-style-type: none"> min. : 0,14 mm² (AWG 25) max. : 1,5 mm² (AWG 15) 0,45 Nm (4 lb-in)
Séparation galvanique <ul style="list-style-type: none"> Interface PROFIBUS à la terre Interface PROFIBUS au circuit interne 	710 V CC pendant une minute

Caractéristiques techniques	
Dimensions et poids	
• Largeur	• 30 mm
• Hauteur	• 100 mm
• Profondeur	• 75 mm
Poids	
• Poids net	• 134 g
• Poids avec l'emballage	• 171 g

*) La charge actuelle d'un client externe connecté entre VP (broche 6) et DGND (broche 5) ne doit pas dépasser un maximum de 15 mA (résistant au court-circuit) pour la terminaison de bus.

Remarque

Le CM 1243- (module maître PROFIBUS) doit être alimenté en courant par l'alimentation capteur 24 V CC de la CPU.

A.9.1.4 Le maître PROFIBUS (CM 1243-5) reçoit son alimentation 24 V- de la CPU.

Remarque

Le CM 1243-5 (module maître PROFIBUS) doit être alimenté en courant par l'alimentation capteur 24 V- de la CPU.

A.9.1.5 Brochage de la prise subD du CM 1243-5

Interface PROFIBUS

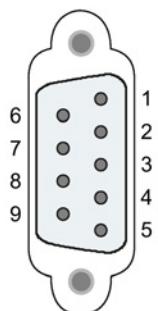


Tableau A- 63Brochage de la prise subD

Broche	Description	Broche	Description
1	- non utilisée -	6	VP : Alimentation +5 V uniquement pour les résistances de terminaison de bus ; pas pour alimenter des périphériques externes
2	- non utilisée -	7	- non utilisée -
3	RxD/TxD-P : Ligne de données B	8	RxD/TxD-N : Ligne de données A
4	CNTR-P : RTS	9	- non utilisée -
5	DGND : Terre pour les signaux de données et VP	Boîtier	Raccord de mise à la terre

Câble PROFIBUS

Remarque

Mise en contact du blindage du câble PROFIBUS

Le blindage du câble PROFIBUS doit être mis en contact.

Pour ce faire, dénudez l'isolation de l'extrémité du câble PROFIBUS et reliez le blindage à la terre fonctionnelle.

A.9.2 CP GPRS**Remarque****Le CP 1242-7 n'est pas homologué pour le domaine maritime**

Le CP 1242-7 ne dispose pas de l'homologation pour le domaine maritime :

Remarque

Le firmware de votre CPU doit être de version V2.0 ou plus pour que vous puissiez utiliser ces modules.

A.9.2.1 CP 1242-7 GPRS

Tableau A- 64Caractéristiques techniques du CP 1242-7 GPRS V2

Caractéristiques techniques	
Référence	6GK7 242-7KX3-0XE0
Interface sans fil	
Connecteur d'antenne	Prise SMA
Impédance nominale	50 ohms
Liaison sans fil	
Puissance d'émission maximum	<ul style="list-style-type: none"> • GSM 850, classe 4 : +33 dBm ±2dBm • GSM 900, classe 4 : +33 dBm ±2dBm • GSM 1800, classe 1 : +30 dBm ±2dBm • GSM 1900, classe 1 : +30 dBm ±2dBm
GPRS	Multi-emplacement classe 10 classe d'appareil B schéma de codage 1...4 (GMSK)
SMS	Mode disparaisant : MO service : point à point
Conditions ambiantes autorisées	
Température ambiante	<ul style="list-style-type: none"> • pendant le stockage • pendant le transport • pendant le fonctionnement avec une installation verticale (profilé support horizontal) • pendant le fonctionnement avec une installation horizontale (profilé support vertical) <ul style="list-style-type: none"> • -40 °C à 70 °C • 0 °C à 55 °C • 0 °C à 45 °C
Humidité relative à 25 °C durant le fonctionnement, sans condensation, maximum	95 %
Degré de protection	IP20
Alimentation, consommation de courant et dissipation de courant	
Type d'alimentation	Courant continu

Caractéristiques techniques	
Alimentation électrique / externe	24 V • minimum • maximum
Consommation de courant (typique)	• à partir de 24 V CC • depuis le bus interne du S7-1200
Perte de puissance effective (typique)	• à partir de 24 V CC • depuis le bus interne du S7-1200
Alimentation 24 V CC	• Section du câble min. • Section du câble max. • Couple de serrage des bornes à vis
Séparation galvanique	710 V CC pendant une minute
Unité d'alimentation électrique au circuit interne	
Dimensions et poids	
• Largeur • Hauteur • Profondeur	• 30 mm • 100 mm • 75 mm
Poids	• Poids net • Poids avec l'emballage
	• 133 g • 170 g

A.9.2.2 Antenne GSM/GPRS ANT794-4MR

Caractéristiques techniques de l'antenne GSM/GPRS ANT794-4MR

ANT794-4MR	
Référence	6NH9860-1AA00
Réseaux mobiles sans fil	GSM/GPRS
Plages de fréquence	• 824 à 960 MHz (GSM 850, 900) • 1 710 à 1 880 MHz (GSM 1 800) • 1 900 à 2 200 MHz (GSM / UMTS)
Particularités	omnidirectionnel
Gain d'antenne	0 dB
Impédance	50 ohms
Rapport d'ondes stationnaires (SWR)	< 2,0
Puissance max.	20 W
Polarité	linéaire verticale

Caractéristiques techniques

A.9 Interfaces de communication

ANT794-4MR	
Connecteur	SMA
Longueur du câble d'antenne	5 m
Matériau extérieur	PVC dur, résistant aux UV
Degré de protection	IP20
Conditions ambiantes autorisées	<ul style="list-style-type: none">• Température de fonctionnement• Température de transport/stockage• Humidité relative <ul style="list-style-type: none">• -40 °C à +70 °C• -40 °C à +70 °C• 100 %
Matériau extérieur	PVC dur, résistant aux UV
Construction	Antenne avec câble fixe de 5 m et connecteur mâle SMA
Dimensions (P x H) en mm	25 x 193
Poids	<ul style="list-style-type: none">• Antenne comprenant un câble• Raccords <ul style="list-style-type: none">• 310 g• 54 g
Installation	Avec support fourni

A.9.2.3 Antenne en nappe ANT794-3M

Caractéristiques techniques de l'antenne en nappe ANT794-3M

ANT794-3M		
Référence	6NH9870-1AA00	
Réseaux mobiles sans fil	GSM 900	GSM 1800/1900
Plages de fréquence	890 - 960 MHz	1710 - 1990 MHz
Rapport d'ondes stationnaires (VSWR)	≤ 2:1	≤ 1,5:1
Affaiblissement d'adaptation (Tx)	≈ 10 dB	≈ 14 dB
Gain d'antenne	0 dB	
Impédance	50 ohms	
Puissance max.	10 W	
Câble d'antenne	Câble HF RG 174 (fixe) avec connecteur mâle SMA	
Longueur de câble	1,2 m	
Degré de protection	IP64	
Plage de températures autorisée	-40°C à +75°C	
Inflammabilité	UL 94 V2	
Matériau extérieur	ABS Polylac PA-765, gris clair (RAL 7035)	
Dimensions (l x L x H) en mm	70,5 x 146,5 x 20,5	
Poids	130 g	

A.9.3**Teleservice (TS)**

Les manuels suivants contiennent les caractéristiques techniques des adaptateurs TS Adapter IE Basic et TS Adapter modulaire :

- Industrial Software Engineering Tools
TS Adapter modulaire
- Industrial Software Engineering Tools
TS Adapter IE Basic

Pour plus d'informations sur ce produit et pour obtenir la documentation du produit, reportez-vous au site Web du catalogue produit pour l'adaptateur TS Adapter

(<https://eb.automation.siemens.com/mall/en/de/Catalog/Search?searchTerm=TS%20Adapter%20IE%20basic&tab=>).

A.9.4 Communication RS485, RS232 et RS422**A.9.4.1 Caractéristiques CB 1241 RS485****Remarque**

Le firmware de votre CPU doit être de version V2.0 ou plus pour que vous puissiez utiliser ce CB.

Tableau A- 65Caractéristiques générales

Caractéristiques techniques	CB 1241 RS485
Référence	6ES7 241-1CH30-1XB0
Dimensions L x H x P (mm)	38 x 62 x 21
Poids	40 grammes

Tableau A- 66Emetteur et récepteur

Caractéristiques techniques	CB 1241 RS485
Type	RS485 (semi-duplex 2 fils)
Plage de tension en mode commun	-7 V à +12 V, 1 seconde, 3 V eff. continu
Tension de sortie différentielle émetteur	2 V min. pour $R_L = 100 \Omega$ 1,5 V min. pour $R_L = 54 \Omega$
Terminaison et polarisation	10K vers +5 V sur B, RS485 broche 3 10K vers GND sur A, RS485 broche 4
Terminaison optionnelle	Court-circuiter broche TB sur broche T/RB, impédance de terminaison effective 127 Ω , connexion à RS485 broche 3 Court-circuiter broche TA sur broche T/RA, impédance de terminaison effective 127 Ω , connexion à RS485 broche 4
Impédance d'entrée de récepteur	5,4 k Ω min., terminaison incluse
Seuil/sensibilité récepteur	+/- 0,2 V min., hystérésis typique 60 mV
Isolation	500 V CA, 1 minute
signal RS485 à la terre du châssis	
signal RS485 au commun logique CPU	
Longueur de câble, blindé	1000 m max.
Débit en bauds	300 bauds, 600 bauds, 1,2 kbits, 2,4 kbits, 4,8 kbits, 9,6 kbits (par défaut), 19,2 kbits, 38,4 kbits, 57,6 kbits, 76,8 kbits, 115,2 kbits
Parité	Sans parité (par défaut), parité paire, parité impaire, parité marque (bit de parité toujours à 1), parité espace (bit de parité toujours à 0)
Nombre de bits d'arrêt	1 (par défaut), 2
Contrôle du flux	Non pris en charge
Temps d'attente	0 à 65 535 ms

Tableau A- 67Alimentation

Caractéristiques techniques	CB 1241 RS485
Perte de puissance (dissipation)	1,5 W
Consommation de courant (bus SM), max.	50 mA
Consommation de courant (24 V-) max.	80 mA

CB 1241 RS485 (6ES7 241-1CH30-1XB0)	
<p>① Connectez "TA" et TB" comme indiqué pour terminer le réseau. La terminaison ne concerne que les appareils aux extrémités du réseau RS485.</p> <p>② Utilisez un câble à paire torsadée blindée et connectez le blindage du câble à la terre.</p>	

La terminaison ne concerne que les deux extrémités du réseau RS485. Il n'y a pas de terminaison ni de polarisation des appareils situés entre les deux appareils d'extrémité. Voir le chapitre "Polarisation et terminaison d'un connecteur de réseau RS485".

Tableau A- 68Brochage pour le CB 1241 RS485 (6ES7 241-1CH30-1XB0)

Broche	Connecteur à 9 broches	X20
1	RS485 / Terre logique	--
2	RS485 / Non utilisé	--
3	RS485 / TxD+	3 - T/RB
4	RS485 / RTS	1 - RTS
5	RS485 / Terre logique	--
6	RS485 / Alim. 5 V	--
7	RS485 / Non utilisé	--

Caractéristiques techniques

A.9 Interfaces de communication

Broche	Connecteur à 9 broches	X20
8	RS485 / TxD-	4 - T/RA
9	RS485 / Non utilisé	--
Boîtier		7 - M

A.9.4.2 Caractéristiques du module CM 1241 RS422/485

Caractéristiques du module CM 1241 RS422/485

Tableau A- 69Caractéristiques générales

Caractéristiques techniques	CM 1241 RS422/485
Référence	6ES7 241-1CH32-0XB0
Dimensions L x H x P (mm)	30 x 100 x 75
Poids	155 grammes

Tableau A- 70Emetteur et récepteur

Caractéristiques techniques	CM 1241 RS422/485
Type	RS422 ou RS485, connecteur femelle subD 9 broches
Plage de tension en mode commun	-7 V à + 12 V, 1 seconde, 3 V eff. continu
Tension de sortie différentielle émetteur	2 V min. pour $R_L = 100 \Omega$ 1,5 V min. pour $R_L = 54 \Omega$
Terminaison et polarisation	10 k Ω vers + 5 V sur B, PROFIBUS broche 3 10 k Ω vers GND sur A, PROFIBUS broche 8 Options de polarisation interne fournies ou pas de polarisation interne. Une terminaison externe est nécessaire dans tous les cas, voir Polarisation et terminaison d'un connecteur réseau RS485 et Configuration de RS422 et RS485 dans le Manuel système de l'automate programmable S7-1200.
Impédance d'entrée de récepteur	5,4 k Ω min., terminaison incluse
Seuil/sensibilité récepteur	+/- 0,2 V min., hystérésis typique 60 mV
Isolation	500 V CA, 1 minute
signal RS485 à la terre du châssis	
signal RS485 au commun logique CPU	
Longueur de câble, blindé	1000 m max. (en fonction du débit)
Débit en bauds	300 bauds, 600 bauds, 1,2 kbits, 2,4 kbits, 4,8 kbits, 9,6 kbits (par défaut), 19,2 kbits, 38,4 kbits, 57,6 kbits, 76,8 kbits, 115,2 kbits
Parité	Sans parité (par défaut), parité paire, parité impaire, parité marque (bit de parité toujours à 1), parité espace (bit de parité toujours à 0)
Nombre de bits d'arrêt	1 (par défaut), 2
Contrôle du flux	XON/XOFF pris en charge pour le mode RS422
Temps d'attente	0 à 65 535 ms

Tableau A- 71Alimentation

Caractéristiques techniques	CM 1241 RS422/485
Perte de puissance (dissipation)	1,1 W
De + 5 V-	220 mA

Tableau A- 72Connecteur RS485 ou RS422 (femelle)

Broche	Description	Connecteur (femelle)	Broche	Description
1	Terre logique ou communication		6 PWR	+ 5 V avec résistance série 100 ohms : sortie
2 TxD+ ¹	Connecté pour RS422 Non utilisé pour RS485 : sortie		7	Non connecté
3 TxD+	Signal B (RxD/TxD+) : entrée/sortie		8 TXD-	Signal A (RxD/TxD-) : entrée/sortie
4 RTS ²	Demande pour émettre (niveau TTL) : sortie		9 TXD- ¹	Connecté pour RS422 Non utilisé pour RS485 : sortie
5 GND	Terre logique ou communication		Boîtier	Terre du châssis

- ¹ Les broches 2 et 9 sont utilisées uniquement comme signaux d'émission pour RS422.
² RTS est un signal de niveau TTL et peut être utilisé pour piloter un autre appareil semi-duplex en fonction de ce signal. Il est actif lorsque vous émettez et inactif le reste du temps.

A.9.4.3 Caractéristiques du CM 1241 RS232

Tableau A- 73Caractéristiques générales

Caractéristiques techniques	CM 1241 RS232
Référence	6ES7 241-1AH32-0XB0
Dimensions (mm)	30 x 100 x 75
Poids	150 grammes

Tableau A- 74Emetteur et récepteur

Caractéristiques techniques	CM 1241 RS232
Type	RS232 (duplex intégral)
Tension de sortie émetteur	+/- 5 V min. pour $R_L = 3 \text{ k}\Omega$
Tension de sortie émetteur	+/- 15 V CC max.
Impédance d'entrée de récepteur	3 $\text{k}\Omega$ min.
Seuil/sensibilité récepteur	0,8 V min. bas, 2,4 V max. haut, 0,5 V hystérésis typique
Tension d'entrée récepteur	+/- 30 VDC max.

Caractéristiques techniques

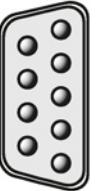
A.10 Modules technologiques

Caractéristiques techniques	CM 1241 RS232
Isolation signal RS232 à la terre du châssis signal RS232 au commun logique CPU	500 V CA, 1 minute
Longueur de câble, blindé	10 m max.
Débit	300 bauds, 600 bauds, 1,2 kbits, 2,4 kbits, 4,8 kbits, 9,6 kbits (par défaut), 19,2 kbits, 38,4 kbits, 57,6 kbits, 76,8 kbits, 115,2 kbits
Parité	Sans parité (par défaut), parité paire, parité impaire, parité marque (bit de parité toujours à 1), parité espace (bit de parité toujours à 0)
Nombre de bits d'arrêt	1 (par défaut), 2
Contrôle du flux	Matériel, logiciel
Temps d'attente	0 à 65 535 ms

Tableau A- 75Alimentation

Caractéristiques techniques	CM 1241 RS232
Perte de puissance (dissipation)	1 W
De + 5 V-	200 mA

Tableau A- 76Connecteur RS232 (mâle)

Broche	Description	Connecteur (mâle)	Broche	Description
1 DCD	Détection de porteuse : Entrée		6 DSR	Modem prêt : Entrée
2 RxD	Données reçues de DCE : Entrée		7 RTS	Demande pour émettre : Sortie
3 TxD	Données émises vers DCE : Sortie		8 CTS	Prêt à émettre : Entrée
4 DTR	Terminal de données prêt : Sortie		9 RI	Indicateur d'appel (inutilisé)
5 GND	Terre logique		Boîtier	Terre du châssis

A.10 Modules technologiques

A.10.1 SM Maître SM 1278 4xIO-Link

A.10.1.1 Caractéristiques techniques du module d'entrées-sorties du Maître SM 1278 4xIO-Link

Tableau A- 77Caractéristiques générales

Caractéristiques techniques		Module d'entrées-sorties du Maître SM 1278 4xIO-Link
Référence		6ES7 278-4BD32-0XB0
Dimensions L x H x P (mm)		45 x 100 x 75
Poids		150 grammes
Informations générales		
	données I&M	Oui ; IM0 à IM3
Tension d'alimentation		
	Tension nominale (CC)	24 V-
	Limite inférieure de plage valide (CC)	19,2 V ; 20,5 V si IO-Link est utilisé (la tension d'alimentation pour les appareils IO-Link sur le maître doit être au moins de 20 V)
	Limite supérieure de plage valide (CC)	28,8 V-
	Protection contre l'inversion de polarité	Oui
Courant d'entrée		
	Consommation de courant	65 mA ; sans charge
Alimentation du codeur		
	Nombre de sorties	4
	Courant de sortie, valeur nominale	200 mA
Perte de puissance		
	Perte de puissance, typ.	1 W, à l'exclusion du chargement de port
Entrées/sorties TOR		
	Longueur de câble (mètres)	20 m, non blindé, max.
SDLC		
	Longueur de câble (mètres)	20 m, non blindé, max.
IO-Link		
	Nombre de ports	4
	Nombre de ports qui peuvent être commandés en même temps	4
	protocole 1.0 IO-Link	Oui
	protocole 1.1 IO-Link	Oui
Etat de fonctionnement		
	IO-Link	Oui
	DI	Oui
	DQ	Oui ; max. 100 mA

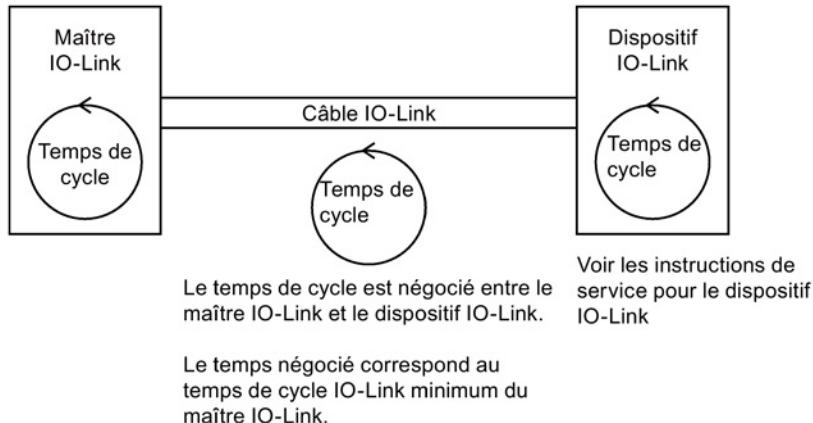
Caractéristiques techniques

A. 10 Modules technologiques

Caractéristiques techniques		Module d'entrées-sorties du Maître SM 1278 4xIO-Link
Raccordement d'appareils IO-Link		
Port type A		Oui
Vitesse de transmission		4,8 kBd (COM1) 38,4 kBd (COM2) 230,4 kBd (COM3)
Temps de cycle, min.		2 ms ; dynamique, dépend de la longueur des données de l'utilisateur
Taille des données de traitement, entrée par port		32 octets ; max.
Taille des données de traitement, entrée par module		32 octets
Taille des données de traitement, sortie par port		32 octets ; max.
Taille des données de traitement, sortie par module		32 octets
Taille mémoire pour les paramètres d'appareil		2 Ko
Longueur de câble non blindé, max. (mètres)		20 m
Informations sur les alarmes/les diagnostics/l'état		
Affichage de l'état		Oui
Alarmes		
Alarme de diagnostic		Oui ; le diagnostic de port n'est disponible que dans le mode IO-Link
Alarmes de diagnostic		
Diagnostic		
Surveillance de la tension d'alimentation		Oui
Court-circuit		Oui
LED de signalisation de diagnostic		
Surveillance de la tension d'alimentation		Oui ; DEL DIAG rouge clignotante
Affichage d'état de canal		Oui ; une DEL verte par canal pour l'état de canal Qn (état SIO) et l'état de PORT Cn (état IO-Link)
Pour le diagnostic de canal		Oui ; DEL Fn rouge
Pour le diagnostic de modules		Oui ; DEL DIAG verte/rouge
Séparation galvanique		
Canaux de séparation galvanique		
Entre les canaux		Non
Entre les canaux et le bus du fond de panier		Oui
Écart de potentiel autorisé		
Entre les circuits différents		75 V CC / 60 V CA (isolation de base)
Isolation		
Isolation testée avec		707 V CC (test de type)
Conditions ambiantes		
Température de fonctionnement		
Min.		- 20 °C
Débit		60 °C
Montage horizontal, min.		-20 °C
Montage horizontal, max.		60 °C
Montage vertical, min.		-20 °C

Caractéristiques techniques	Module d'entrées-sorties du Maître SM 1278 4xIO-Link
Montage vertical, max.	50 °C

Présentation générale du temps de réponse



A.10.1.2 Schémas de câblage du module SM 1278 4 x IO-Link maître

Tableau A- 78Schéma de câblage du module SM 1278 IO-Link maître

SM 1278 IO-Link maître (6ES7 278-4BD32-0XB0)

Tableau A- 79 Brochage pour le SM 1278 IO-Link maître (6ES7 278-4BD32-0XB0)

Broche	X10	X11	X12	X13
1	L+ / 24 V CC	Pas de connexion	Pas de connexion	Pas de connexion
2	M / 24 V CC	Pas de connexion	Pas de connexion	Pas de connexion
3	Terre fonctionnelle	Pas de connexion	Pas de connexion	Pas de connexion
4	Pas de connexion	Pas de connexion	Pas de connexion	Pas de connexion
5	L ₁	L ₂	L ₃	L ₄
6	C/Q ₁	C/QL ₂	C/Q ₃	C/QL ₄
7	ML ₁	ML ₂	M ₃	ML ₄

A.11 Produits connexes

A.11.1 Module d'alimentation PM 1207

Le PM 1207 est un module d'alimentation pour le SIMATIC S7-1200. Il présente les caractéristiques suivantes :

- Entrée 120/230 V CA, sortie 24 V-/2,5 A
- Référence 6ESP 332-1SH71-4AA0

Pour plus d'informations sur ce produit et pour obtenir la documentation du produit, recherchez PM 1207 dans le catalogue de produits en ligne (<https://eb.automation.siemens.com/mall/en/de/Catalog/Product/6AG1332-1SH71-4AA0>).

A.11.2 Module commutateur compact CSM 1277

Le CSM1277 est un module commutateur compact Industrial Ethernet. Il peut être utilisé pour multiplier l'interface Ethernet du S7-1200 pour permettre des communications simultanées avec les pupitres opérateurs, les consoles de programmation et autres contrôleurs. Il présente les caractéristiques suivantes :

- 4 prises RJ45 pour connexion à l'Industrial Ethernet
- 3 prises polarisées dans le bornier pour connexion à l'alimentation externe 24 V CC sur le dessus
- Des DEL pour l'affichage de diagnostic et d'état des ports Industrial Ethernet
- Référence 6GK7 277-1AA00-0AA0

Pour plus d'informations sur ce produit et pour obtenir la documentation du produit, recherchez CSM 1277 dans le catalogue de produits en ligne

(<https://eb.automation.siemens.com/mall/en/de/Catalog/Search?searchTerm=csm%201277&tab=>).

A.11.3 Module CM CANopen

Le module CM CANopen est un module enfichable entre l'API SIMATIC S7-1200 et tout appareil compatible CANopen. Le CM CANopen peut être configuré pour être aussi bien maître qu'esclave. Il y a deux CM CANopen modules : le module CANopen (référence 021620-B) et le module CANopen (Ruggedized) (référence 021730-B).

Le module CANopen présente les caractéristiques suivantes :

- Capable de connecter 3 modules par CPU
- Connecte jusqu'à 16 abonnés esclaves CANopen
- 256 octets d'entrée et 256 octets de sortie par module
- 3 LED fournissent des informations de diagnostic sur le module, le réseau et l'état d'E/S
- Prend en charge le stockage de la configuration de réseau CANopen dans l'API
- Le module peut être intégré dans le catalogue du matériel de la suite de configuration TIA Portal
- Configuration CANopen avec CANopen Configuration Studio (inclus) ou avec tout autre outil de configuration CANopen externe
- Est conforme aux profils de communication CANopen CiA 301 rev. 4.2 et CiA 302 rev. 4.1
- Prend en charge CAN 2.0A pour la gestion de protocoles personnalisés
- Blocs fonctionnels préprogrammés disponibles pour la programmation chaque API dans TIA Portal
- Les modules CM CANopen comprennent des connecteurs Sub-D avec bornes à vis pour sous-réseau, un CD CM CANopen Configuration Studio et un câble de configuration USB

Pour plus d'informations sur ce produit et pour obtenir la documentation du produit, recherchez CM CANopen dans le catalogue de produits en ligne.

Remplacement d'une CPU V3.0 par une CPU V4.1

B.1

Remplacement d'une CPU V3.0 par une CPU V4.1

Vous pouvez remplacer votre CPU V3.0 par une CPU V4.1 (Page 79) et utiliser le projet STEP 7 existant que vous avez conçu pour la CPU V3.0. Vous ne pouvez pas mettre à niveau une CPU V3.0 en CPU V4.1 à travers une mise à jour du firmware ; vous devez remplacer le matériel. Lorsque vous remplacez une CPU V3.0 par une CPU V4.1, vous pourriez également vérifier s'il y a des mises à jour du firmware (Page 368) et les appliquer à vos modules d'entrées-sorties et de communication connectés.

Remarque

Impossible de remplacer un appareil V4.1 par un appareil V3.0

Vous pouvez remplacer une CPU V3.0 par une CPU V4.1, mais il n'est pas possible de remplacer une CPU V4.1 par une CPU V3.0 après avoir téléchargé la configuration. Si vous souhaitez visualiser ou utiliser votre projet STEP 7 V3.0 existant, créez une archive de votre projet STEP 7 V3.0 avant de changer l'appareil.

Notez que si vous n'avez pas téléchargé la configuration de l'appareil remplacé, vous pouvez annuler. Cependant, une fois la configuration téléchargée, vous ne pouvez pas annuler le remplacement de la V3.0 par la V4.1.

Vous devez tenir compte de certains changements dans la configuration et le fonctionnement entre les deux versions de CPU :

Blocs d'organisation

Avec la V4.1, vous pouvez configurer l'exécution de l'OB pour qu'elle soit interruptible ou non interruptible (Page 58). Pour les projets d'anciennes CPU V3.0, STEP 7 définit tous les OB pour qu'ils soient non interruptibles par défaut.

STEP 7 règle toutes les priorités de l'OB (Page 58) sur les mêmes valeurs qu'elles avaient dans le projet STEP 7 CPU V3.0.

Si vous le souhaitez, vous pouvez par la suite modifier les paramètres de l'interruptibilité ou de priorité.

Les informations de déclenchement de l'OB d'alarme de diagnostic se réfèrent à l'ensemble du sous-module si aucun événement de diagnostic n'est en attente.

Protection par mot de passe de la CPU

STEP 7 définit le niveau de protection par mot de passe (Page 89) pour la CPU V4.1 de sorte qu'il soit équivalent à celui défini pour la CPU V3.0, et affecte le mot de passe V3.0 au mot de passe "Accès intégral (aucune protection)" pour la CPU V4.1 :

Niveau de protection V3.0	Niveau de protection V4.1
Pas de protection	Accès illimité (aucune protection)
Protection en écriture	Accès en lecture
Protection en écriture / lecture	Accès IHM

Notez que le niveau d'accès "Aucun accès (protection intégrale)" V4.1 n'existe pas pour la V3.0.

Serveur Web

Si vous utilisez les pages Web personnalisées dans votre projet V3.0, sauvegardez-les dans votre dossier d'installation du projet dans le sous-dossier "UserFiles\Webserver" avant de mettre à niveau votre projet. Si vous sauvegardez vos pages personnalisées à cet emplacement, enregistrer le projet STEP 7 enregistrera également les pages Web personnalisées.

Si vous remplacez une CPU V3.0 par une CPU V4.1, votre paramètre serveur Web du projet pour l'activation du serveur Web sera le même que dans la V3.0. Vous pouvez alors configurer les utilisateurs, priviléges, mots de passe (Page 263), et les langues comme vous en avez besoin pour utiliser le Serveur Web. Si vous ne conférez pas aux utilisateurs de droits supplémentaires, vous ne verrez que les pages Web standard (Page 264). La CPU S7-1200 V4.1 ne prend pas en charge l'utilisateur et le mot de passe "admin" préalablement préconfigurés.

Auparavant, la page Data log Web server du S7-1200 V3.0 permettait une opération "Télécharger et Effacer". La page File browser (Page 264) du serveur Web V4.1, à partir de laquelle vous pouvez accéder aux journaux de données, ne permet plus cette fonction. A la place, le serveur Web offre la possibilité de télécharger, renommer et d'effacer les fichiers journaux.

Incompatibilité de la carte de transfert

Vous ne pouvez pas utiliser de carte de transfert (Page 62) V3.0 pour transférer un programme V3.0 vers une CPU V4.1. Vous devez ouvrir le projet V3.0 dans STEP 7, remplacer l'appareil par une CPU V4.1 (Page 79) et charger le projet STEP 7 sur votre CPU V4.1. Une fois que vous avez changé votre projet en projet V4.1, vous pouvez ensuite créer une carte de transfert V4.1 pour les transferts de programmes suivants.

Communication GET/PUT

La communication GET/PUT est activée par défaut dans la version V3.0. Lorsque vous remplacez votre CPU V3.0 par une CPU V4.1 (Page 79), un message apparaît dans la section informations de compatibilité et indique que GET/PUT est activé.

Prise en charge de Motion Control

Les CPU S7-1200 V4.1 ne prennent pas en charge les bibliothèques Motion Control V1.0 et V2.0. Si vous changez d'appareil pour un projet STEP 7 avec des bibliothèques Motion Control V1.0 ou V2.0, le changement d'appareil entraîne le remplacement des instructions de la bibliothèque Motion Control V1.0 ou V2.0 par les instructions Motion Control (Page 324) V3.0 compatibles à la compilation.

Si vous remplacez une CPU V3.0 par une CPU V4.1 pour un projet STEP 7 qui contient deux versions différentes d'instructions Motion Control (V3.0 et V5.0), le changement d'appareil entraîne le remplacement des instructions Motion Control (Page 324) compatibles avec V5.0 à la compilation.

Pendant le remplacement d'une CPU V3.0 par une CPU V4.1, la version de l'objet technologique (TO) Motion control ne passe pas automatiquement de la version V3.0 à la version V5.0. Si vous voulez effectuer une mise à niveau aux dernières versions, vous devez aller dans l'arborescence Instructions et sélectionner la version requise de S7-1200 Motion Control pour votre projet, comme indiqué dans le tableau ci-dessous :

Version de la CPU	Versions autorisées de Motion control
V4.1 (Motion control V5.0)	V5.0 ou V4.0 ou V3.0
V4.0 (Motion control V4.0)	V4.0 ou V3.0
V3.0 (Motion control V3.0)	V3.0

La structure du TO est différente entre la version V3.0 et la version V5.0 de Motion Control. Tous les blocs associés sont également associés. Les interfaces de bloc, les tables de visualisation et les traces sont mises à jour vers la nouvelle structure de Motion control V5.0. Vous trouverez les différences entre les paramètres d'axe Motion Control CPU V3.0 et CPU V4.1 dans les deux tableaux suivants :

CPU V3.0 (Motion control V3.0)	CPU V4.1 (Motion control V5.0)
Config.General.LengthUnit	Units.LengthUnit
Config.Mechanics.PulsesPerDriveRevolution	Actor.DriveParameter.PulsesPerDriveRevolution
Config.Mechanics.LeadScrew	Mechanics.LeadScrew
Config.Mechanics.InverseDirection	Actor.InverseDirection
Config.DynamicLimits.MinVelocity	DynamicLimits.MinVelocity
Config.DynamicLimits.MaxVelocity	DynamicLimits.MaxVelocity
Config.DynamicDefaults.Acceleration	DynamicDefaults.Acceleration
Config.DynamicDefaults.Deceleration	DynamicDefaults.Deceleration
Config.DynamicDefaults.EmergencyDeceleration	DynamicDefaults.EmergencyDeceleration
Config.DynamicDefaults.Jerk	DynamicDefaults.Jerk
Config.PositionLimits_SW.Active	PositionLimitsSW.Active
Config.PositionLimits_SW.MinPosition	PositionLimitsSW.MinPosition
Config.PositionLimits_SW.MaxPosition	PositionLimitsSW.MaxPosition
Config.PositionLimits_HW.Active	PositionLimitsHW.Active
Config.PositionLimits_HW.MinSwitchedLevel	PositionLimitsHW.MinSwitchLevel
Config.PositionLimits_HW.MaxSwitchedLevel	PositionLimitsHW.MaxSwitchLevel
Config.Homing.AutoReversal	Homing.AutoReversal

CPU V3.0 (Motion control V3.0)	CPU V4.1 (Motion control V5.0)
Config.Homing.Direction	Homing.ApproachDirection
Config.Homing.SideActiveHoming	Sensor[1].ActiveHoming.SideInput
Config.Homing.SidePassiveHoming	Sensor[1].PassiveHoming.SideInput
Config.Homing.Offset	Sensor[1].ActiveHoming.HomePositionOffset
Config.Homing.FastVelocity	Homing.ApproachVelocity
Config.Homing.SlowVelocity	Homing.ReferencingVelocity
MotionStatus.Position	Position
MotionStatus.Velocity	Velocity
MotionStatus.Distance	StatusPositioning.Distance
MotionStatus.TargetPosition	StatusPositioning.TargetPosition
StatusBits.SpeedCommand	StatusBits.VelocityCommand
StatusBits.Homing	StatusBits.HomingCommand

Le seul paramètre "commandtable" renommé est le tableau avec les commandes :

V3.0	V4.1
Config.Command[]	Command[]

Remarque : Le tableau "Command[]" est un type de données utilisateur du type "TO_CmdTab_Config_Command" dans la V3.0 et "TO_Struct_Command" dans la V4.1.

Modifications des instructions

Les instructions suivantes ont subi des modifications de paramètres ou de comportement :

- RDREC et WRREC (Page 154)
- CONV (Page 115)

Communication de pupitres IHM

Si vous aviez un ou plusieurs pupitres IHM (Page 20) connectés à votre CPU S7-1200 V3.0, la communication à la CPU S7-1200 V4.1 dépend du type de communication utilisé et de la version de firmware du pupitre IHM. Recompilez puis téléchargez le projet sur la CPU et l'IHM et/ou mettez à jour le firmware IHM.

Conditions pour recompiler des blocs de programme

Après avoir remplacé une CPU V3.0 par une CPU V4.1, vous devez recompiler tous les blocs de programme avant de pouvoir les charger sur la CPU V4.1. De plus, si des blocs ont une protection know-how (Page 92) ou une protection contre la copie liée à un numéro de série API (Page 93), vous devez retirer la protection avant de compiler et charger les blocs. (En revanche, il n'est pas nécessaire de désactiver la protection contre la copie liée à une carte mémoire.) Après une compilation réussie, vous pouvez reconfigurer la protection know-how et/ou la protection contre la copie du numéro de série API. Notez que si votre projet inclut des blocs avec une protection know-how fournis par un OEM, vous devez contacter l'OEM pour qu'il fournisse les versions V4.1 de ces blocs.

En général, Siemens vous recommande de recompiler la configuration matérielle et le logiciel dans STEP 7 et de les charger sur tous les appareils dans votre projet après le changement d'appareil. Corrigez toutes les erreurs rencontrées lors de la compilation du projet, puis recompilez jusqu'à ce qu'il n'y ait plus d'erreur. Ensuite, vous pouvez charger le projet sur la CPU V4.1.

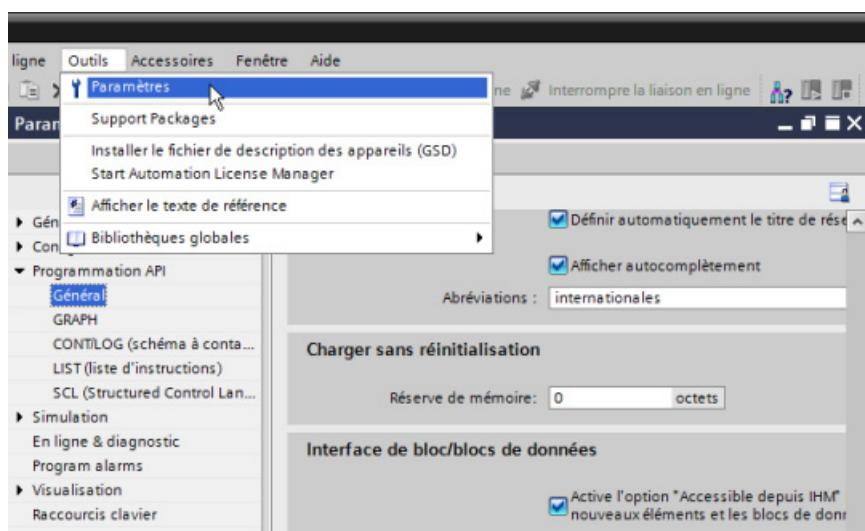
Les projets S7-1200 V3.0 peuvent ne pas avoir suffisamment d'espace dans les CPU S7-1200 V4.1

S7-1200 V4.0 a ajouté un espace de réserve de 100 octets à chaque DB pour prendre en charge le chargement sans réinitialisation.

Vous pouvez supprimer la zone de réserve de 100 octets des DB avant d'essayer de télécharger un projet V3.0 vers une CPU V4.1.

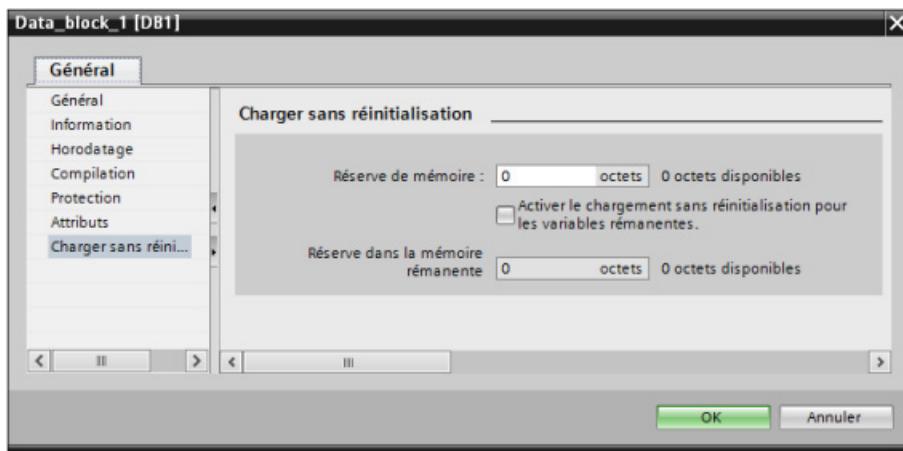
Pour supprimer l'espace de réserve de 100 octets, suivez les étapes suivantes avant de procéder au remplacement d'un appareil :

1. à partir du menu principal du Portail TIA, sélectionnez la commande de menu Outils > Paramètres.
2. Dans l'arbre de navigation, ouvrez le nœud Programmation API > Général.
3. Dans la rubrique "Charger sans réinitialisation", réglez la réserve de mémoire sur 0 octets.



Si vous avez déjà effectué le remplacement de l'appareil, vous devez supprimer la réserve de 100 octets dans chaque bloc individuellement :

1. à partir de l'arbre du projet, effectuez un clic droit avec la souris sur un bloc de données à partir du dossier Blocs de programme et sélectionnez Propriétés à partir du menu contextuel.
2. Dans la boîte de dialogue des propriétés du bloc de données, sélectionnez le nœud "Charger sans réinitialisation".
3. Réglez la réserve de mémoire sur 0 octets.
4. Répétez ces étapes pour chaque bloc de données de votre projet.



Reportez-vous au *manuel système de l'automate programmable S7-1200* pour des détails complets sur les caractéristiques de la version V4.1.

Index

A

- Acquisition des valeurs dans un DB en ligne, 362
- Actualisation de pages Web personnalisées, 270
- Actualisation des pages Web personnalisées, 270
- Adaptateur TS Adapter, 18
- Adressage
 - bloc de données, 66
 - entrées (I) ou sorties (Q) individuelles, 67
 - mémoire globale, 66
 - mémoire image, 66
 - mémoire temporaire, 66
 - Valeurs booléennes ou de bit, 67
 - zones de mémoire, 66
- Adresse IP, 87, 87
 - configuration de la CPU en ligne, 365
- Adresse IP de routeur, 87
- Adresse MAC, 87
- Affichage de l'usage des références croisées, 131
- Ajout d'un type de données à une variable via AT, 71
- Ajouter des entrées ou sorties aux opérations CONT ou LOG, 31
- Ajouter un appareil
 - CPU non spécifiée, 77, 370
 - détection du matériel existant, 77
- Alarmes
 - présentation, 58
 - temps d'attente, 58
- Analyseur logique, 374
- API
 - ajouter des modules, 80
 - appel d'un bloc, 104
 - comparaison et synchronisation, 364
- Configuration HSC, 145
 - copie des blocs depuis une CPU en ligne, 363
 - forçage permanent, 359, 360
- Instructions, 44
- Mise en route, 39
 - présentation de la CPU, 15
 - protection du savoir-faire, 92
 - utilisation de blocs, 95
- Utilisation de blocs, 57
 - variables, 40, 44
 - visualisation, 355
- Appareils IHM
 - liaison réseau, 148
 - présentation, 20

Appel de bloc

- Principes de base, 57

AS-i

- adresse, 174
- ajout d'un esclave AS-i, 173
- ajout d'un module maître AS-i CM 1243-2, 173
- maître AS-i CM 1243-2, 172

Assistance, 5

- Assistance client, 5
- Assistance technique, 5
- Assistance technique Siemens, 5

B

- Barre d'outils Favoris, 31
- Basic panels (HMI), 20
- Battery board, BB 1297, 415
- Bibliothèque du protocole USS, 195
- Bibliothèque globale
 - USS, 195
- Bloc de code, 95
 - alarmes, 17
 - appel d'un bloc, 104
 - blocs d'organisation (OB), 17
 - Blocs d'organisation (OB), 97
 - compteurs (exigences en termes de quantité et de mémoire), 18
 - création, 102
 - DB (bloc de données), 101
 - FB (bloc fonctionnel), 100
 - FC (fonction), 100
 - liaison avec une CPU, une carte mémoire ou un mot de passe, 93
 - mise en route, 39
 - nombre de blocs de code, 17
 - nombre d'OB, 17
 - profondeur d'imbrication, 17
 - protection contre la copie, 93
 - protection du savoir-faire, 92
 - taille du programme utilisateur, 17
 - temporisations (exigences en termes de quantité et de mémoire), 17
 - visualisation, 17
- Bloc de données
 - acquisition des valeurs, 362, 362
 - bloc de données d'instance, 66
 - bloc de données global, 66, 101

- Blocs d'organisation (OB), 97
redéfinition des valeurs initiales, 362, 362
- Bloc de données d'instance, 66
- Bloc de données global, 66, 101
- Bloc de gestion de données (DHB), 101
- Bloc d'organisation
appel, 58
classes de priorité, 58
Configuration du fonctionnement, 99
création, 98
fonction, 58
plusieurs OB de cycle de programme, 98
protection du savoir-faire, 92
traitement, 97, 97
- Bloc fonctionnel (FB)
Bloc de données d'instance, 100
Paramètres de sortie, 100
protection du savoir-faire, 92
Valeur initiale, 100
- Blocs
alarmes, 17, 58
appel d'un autre bloc de code, 104
- Blocs de données (DB), 57
- blocs d'organisation (OB), 17, 58
- Blocs d'organisation (OB), 57, 58
- Blocs fonctionnels (FB), 57
- compteurs (exigences en termes de quantité et de mémoire), 18
- contrôle de cohérence, 132
- copie des blocs depuis une CPU en ligne, 363
- événements, 58
- Fonctions (FC), 57
- mise en route, 102
- nombre de blocs de code, 17
- nombre d'OB, 17, 58
- OB de démarrage, 58
- profondeur d'imbrication, 17
- protection par mot de passe, 92
- taille du programme utilisateur, 17
- temporisations (exigences en termes de quantité et de mémoire), 17
- Types, 57
- visualisation, 17
- Boîte d'opérations
Mise en route, 45
- Boutons MARCHE/ARRET, 33
- C**
- CALCULER (calculer), 117
- mise à l'échelle de valeurs analogiques, 47
- utilisé pour les équations complexes, 46
- Caractéristiques
AI 4 x 13 bit SM 1231, 413
caractéristiques techniques d'ordre général, 381
- CB 1241 RS485, 443
- CM 1241 RS232, 445
- CM 1241 RS422/485, 444
- Compatibilité électromagnétique (CEM), 385
- Conditions ambiantes, 387
- CPU 1211C, 392
- CPU 1212C, 392
- CPU 1214C, 392
- CPU 1215C, 392
- CPU 1217C, 392
- environnements industriels, 384
- homologations, 381
- représentation des entrées analogiques (courant), 418
- représentation des entrées analogiques (tension), 417
- SB 1221 4 DI 24 VDC 200 kHz, 397
- SB 1221 4 DI 5 VDC 200 kHz, 397
- SB 1222 4 DQ 24 VDC 200 kHz, 397
- SB 1222 4 DQ 5 VDC 200 kHz, 397
- SB 1223 2 DI/2 DQ 24 VDC, 397
- SB 1223 DI/DQ 24 VDC 200 kHz, 397
- SB 1223 DI/DQ 5 VDC 200 kHz, 397
- SB 1231 AI 1 x 12 bits, 412
- SB 1231 AI 1 x 16 bits RTD, 424
- SB 1231 AI 1 x 16 bits TC, 424
- SB 1232 AQ 1 x 12 bits, 412
- SM 1221 DI 16 24 VDC, 400
- SM 1221 DI 8 24 VDC, 400
- SM 1222 DQ 16 24 VDC, 401
- SM 1222 DQ 16 Relais, 401
- SM 1222 DQ 8 24 VDC, 401
- SM 1222 DQ 8 Relais, 401
- SM 1222 DQ 8 Relais Inverseur, 401
- SM 1223 DI 120/230 VAC/DQ relais, 404
- SM 1223 DI 16/DQ 16 relais, 403
- SM 1223 DI 8/DQ 8, 403, 403
- SM 1223 DI 8/DQ 8 relais, 403
- SM 1231 AI 4 x 16 bits, 413
- SM 1231 AI 4 x 16 bits TC, 427
- SM 1231 AI 4 x RTD x 16 bits, 425
- SM 1231 AI 8 x 13 bits, 413
- SM 1231 AI 8 x 16 bits TC, 427
- SM 1231 AI 8 x RTD x 16 bits, 425
- SM 1232 AQ 2 x 14 bits, 413
- SM 1232 AQ 4 x 14 bits, 413
- SM 1234 AI 4 x 13 bits/AQ 2 x 14 bits, 413
- temps de réponse transitoire pour entrées, 419
- tensions nominales, 388

- Caractéristiques techniques, 381
 Maître SM 1278 4xIO-Link, 447
 représentation des sorties analogiques (courant), 421
 représentation des sorties analogiques (tension), 421
- Carte mémoire
 Mémoire de chargement, 62
- Carte programme, 62
 CB 1241 RS485, 443
 CEIL (plafonnier), 116
 Changer l'appareil, 79
- Chargement
 découvrir, 370
- Chargement à partir de l'appareil
 copie des blocs depuis une CPU en ligne, 363
 programme utilisateur, 363
- Chargement dans la CPU à l'état MARCHE
 déclenchement depuis STEP 7, 373
 présentation, 371
- Classe de protection, 388
- Colonnes et en-têtes dans les Task Cards, 34
- Commande de configuration (gestion des options), 81
- Commande de mouvement
 configuration de l'axe, 285, 295
 fins de course matériels et logiciels, 314
 MC_ChangeDynamic (modifier les réglages dynamiques pour l'axe), 347
 MC_CommandTable, 344
 MC_Halt (pause de l'axe), 332
 MC_Home (référencer l'axe), 329
 MC_MoveAbsolute (positionnement absolu de l'axe), 334
 MC_MoveJog (déplacer l'axe en mode Manuel à vue), 341
 MC_MoveRelative (positionnement relatif de l'axe), 336
 MC_MoveVelocity (déplacer l'axe à une vitesse prédéfinie), 338
 MC_Power (libérer/bloquer l'axe), 325
 MC_ReadParam (lire les paramètres de l'objet technologique), 351
 MC_Reset (confirmer l'erreur), 328
 MC_WriteParam (écrire dans les paramètres d'un objet technologique), 349
 paramètres de configuration du référencement, 320
 phasage, 281
 présentation, 275
 référencement (séquence pour le référencement actif), 322
 référencement de l'axe, 318
- Communication
 active/passive, 161, 163, 179
 adresse AS-i, 174
 Adresse IP, 87
 adresse PROFIBUS, 170
 configuration, 161, 163, 179
 ID de liaison, 157
 liaison réseau, 148
 nombre de liaisons (PROFINET/PROFIBUS), 152
 paramètres, 161
 PROFINET et PROFIBUS, 149
 Réseau, 147
 TCON_Param, 161
- Communication active/passive
 configuration des partenaires, 163, 179
 ID de liaison, 157
 paramètres, 161
- Communication Board (CB)
 ajouter des modules, 80
 CB 1241 RS485, 443
 DEL de signalisation, 193
 présentation, 19
 RS485, 193
 tableau comparatif, 18
- Communication passive/active
 configuration des partenaires, 163, 179
 ID de liaison, 157
 paramètres, 161
- Communication point à point, 191, 191
- Communication programmable, protocole, 191
- Communication réseau, 147
- Communication S7
 configuration de la liaison, 164
- Communication série, 191
- Communication TCP/IP, 147, 156
- Comparaison et synchronisation des CPU hors ligne/en ligne, 364
- Compatibilité électromagnétique, 386
- Compatibilité électromagnétique (CEM), 385
- Compteur rapide
 Configuration, 145
 fonctionnement, 137
 forçage permanent impossible, 360
 HSC, 135
- Compteurs, 124
 Configuration HSC, 145
 fonctionnement HSC, 137
 HSC (compteur rapide), 135
 quantité, 18
 taille, 18
- Conception d'un système API, 95
- Conception d'un système d'automatisation, 57

- Conditions ambiantes
 conditions de service, 387
 conditions de transport et stockage, 387
- Configuration
 Adresse IP, 87
 adresse PROFIBUS, 170
 ajouter des modules, 80
 AS-i, 174
 découvrir, 77, 370
 HSC (compteur rapide), 145
- Instructions PID_Compact et PID_3Step, 231
- Instructions PID_Temp, 234
- liaison réseau, 148
- pages Web personnalisées, 272
- Paramètres de mise en route, 82
- Port Industrial Ethernet, 87
- PROFIBUS, 170
- PROFINET, 87
- Configuration de l'appareil
 Changer un type d'appareil, 79
- Configuration des appareils, 75
 ajouter des modules, 80
 Ajouter un appareil, 78
 AS-i, 174
 Configuration de la CPU, 82, 86
 Configuration des modules, 82, 86
 découvrir, 77, 370
 liaison réseau, 148
 modules retirés, 38
 port AS-i, 174
 Port Ethernet, 87
 PROFIBUS, 170
 PROFINET, 87
- Configuration des paramètres
 CPU, 82, 86
 modules, 82, 86
 Port Ethernet, 87
 PROFINET, 87
- Configuration matérielle, 75
 ajouter des modules, 80
 Ajouter un appareil, 78
 AS-i, 174
 Configuration de la CPU, 82, 86
 Configuration des modules, 82, 86
 découvrir, 77, 370
 liaison réseau, 148
 port AS-i, 174
 Port Ethernet, 87
 PROFIBUS, 170
 PROFINET, 87
- Connecteur de bus, 19
- Connexions
 Liaison IHM, 49
 Liaison réseau, 49
 nombre de liaisons (PROFINET/PROFIBUS), 152
- Connexions maximum du serveur Web, 268
- Conseils
 installation, 23
- Conseils de câblage
 dégagement pour l'écoulement d'air et le refroidissement, 25
- CONT (schéma à contacts)
 éditeur de programme, 356
 état, 356, 359
 présentation, 105
 visualisation, 356
 visualisation de l'état ou de la valeur, 355
- Contacts
 programmation, 42
- Contraintes
 pages Web personnalisées, 271
 serveur Web, 267
- Contrôle de cohérence, 132
- CONV (convertir), 115
- Copie des blocs depuis une CPU en ligne, 363
- Copier valeur (MOVE), instruction, 113
- Copier zone (MOVE_BLK), instruction, 113
- Copier zone contiguë (UMOVE_BLK), instruction, 113
- Courant continu
 sorties, 389
- CPU
 acquisition des valeurs d'un DB, 362
 adresse AS-i, 174
 Adresse IP, 87
 adresse PROFIBUS, 170
 ajouter des modules, 80
 Ajouter un appareil, 78
 appel d'un bloc, 104
 boutons MARCHE/ARRET, 33
 Communication Boards (CB), 19
 comparaison et synchronisation des blocs, 364
 Configuration de la communication avec des IHM, 147
 configuration des appareils, 75
 Configuration des paramètres, 82, 86
 Configuration HSC, 145
 copie des blocs depuis une CPU en ligne, 363
 CPU 1211C, 392
 CPU 1212C, 392
 CPU 1214C, 392
 CPU 1215C, 392
 CPU 1217C, 392
 CPU non spécifiée, 77, 370

- En ligne, 365
 Etats de fonctionnement, 55
 Exécution du programme, 53
 forçage permanent, 359, 360
 liaison réseau, 148
 mémoire tampon de diagnostic, 365
 niveaux de sécurité, 89
 Niveaux de sécurité, 89
 nombre de liaisons de communication, 152
 Paramètres de mise en route, 82
 passage en ligne, 353
 Port Ethernet, 87
 présentation, 15
 PROFINET, 87
 protection d'accès, 89
 protection du savoir-faire, 92
 protection par mot de passe, 89
 pupitre opérateur, 33, 56
 Pupitre opérateur, 354
 redéfinition des valeurs initiales d'un DB, 362
 restauration des réglages d'usine, 366
 Signal Boards (SB), 19
 tableau comparatif, 16
 Tables de visualisation, 357
 temps de réponse transitoire pour entrées analogiques, 419
 Traitement de mise en route, 82
 traitement des OB, 97
 types de communication, 149
 visualisation en ligne, 355
 zone thermique, 22, 25
- CPU non spécifiée, 77, 370, 370
 Création de DB de page Web personnalisée, 272
 Création de pages Web personnalisées, 269
 Création d'une liaison réseau
 entre API, 148
 Crée une liaison IHM, 49
 Crée une liaison réseau
 entre l'API et l'IHM, 49
- Cycle
 forçage permanent, 360, 360
- D**
- Date and Time Long, type de données, 64
 DB (bloc de données), 101
 DB de commande pour les pages Web personnalisées
 paramètre pour l'opération WWW, 273
 DB fragmentaire (pages Web personnalisées)
 génération, 272
 Débogage à l'état MARCHE, 371
- Déclenchement
 traçage, 374
 Découvrir pour télécharger une CPU en ligne, 77
 Dégagement, écoulement d'air et refroidissement, 25
 Degré de protection, 388
 DEL (obtenir état DEL), 133
 DEL de signalisation
 interface de communication, 193
 Opération DEL, 133
 Détection, 370
 Diagnostic
 EtatsAppareil, 134
 EtatsModule, 134
 GET_DIAG, 134
 Get_IM_Data, 134
 indicateur d'état, 85
 tampon, 365
 Diagnostics
 Opération DEL, 133
 Différences
 concernant les instructions point à point, 192
 Documentation, 4
 DTL, type de données, 64
 Durée d'utilisation électrique des relais, 389
- E**
- E/S
 adressage, 68
 forçage permanent, 360
 représentation des entrées analogiques (courant), 418
 représentation des entrées analogiques (tension), 417
 représentation des sorties analogiques (courant), 421
 représentation des sorties analogiques (tension), 421
 Tables de visualisation, 357
 temps de réponse transitoire pour entrées analogiques, 419
 visualisation de l'état dans CONT, 356
- E/S analogiques
 conversion en unités physiques, 47
 représentation des entrées (courant), 418
 représentation des entrées (tension), 417
 représentation des sorties (courant), 421
 représentation des sorties (tension), 421
 temps de réponse transitoire pour entrées, 419
- Ecoulement d'air, 25

Editeur de programmes
 acquisition des valeurs d'un DB, 362
 redéfinition des valeurs initiales d'un DB, 362

Éditeur de programmes
 état, 356
 visualisation, 356

Edition à l'état MARCHE, (Chargement dans la CPU à l'état MARCHE)

Emission, configuration des paramètres, 163, 179

En ligne
 acquisition des valeurs d'un DB, 362

Adresse IP, 365

boutons MARCHE/ARRET, 33

comparaison et synchronisation, 364

découvrir, 370

état, 356

forçage permanent, 359, 360

heure, 365

passage en ligne, 353

pupitre opérateur, 33, 56

Pupitre opérateur, 354

redéfinition des valeurs initiales d'un DB, 362

surveillance de l'état ou de la valeur, 355

Surveillance de l'utilisation de la mémoire, 355

Surveillance du temps de cycle, 354

table de visualisation, 355, 356

Entrée Main, 401

Entrées analogiques
 Spécifications pour CPU, SB et SM, 416
 Spécifications pour RTD/TC SB et SM, 428

Entrées et sorties
 visualisation, 355

Entrées TOR
 Spécifications CA pour SM, 407
 Spécifications V CC pour CPU, SM et SB, 405

Environnements industriels
 homologations, 384

Essai d'isolation à potentiel élevé, 388

Etat
 DEL de signalisation (interface de communication), 193

État
 Opération DEL, 133

Etat ARRET, 55
 boutons de la barre d'outils, 33
 forçage permanent, 360
 Pupitre opérateur, 33, 56, 354

Etat de fonctionnement, 33, 56, 354

État de fonctionnement, 33

Etat MARCHE, 55, 57
 boutons de la barre d'outils, 33
 Exécution du programme, 53

forçage permanent, 360

Pupitre opérateur, 33, 56, 354

Etat MISE EN ROUTE
 Exécution du programme, 53
 forçage permanent, 360

EtatsAppareil, 134

EtatsModule, 134

Ethernet
 Adresse IP, 87
 Communication, 147
 GET, 177
 ID de liaison, 157
 liaison réseau, 148
 mode ad hoc, 157
 Module commutateur compact CSM 1277, 450
 nombre de liaisons de communication, 152
 présentation, 156
 PUT, 177
 types de communication, 149

Événements, 365

Exécute des commandes d'axe en tant que séquence de mouvement (MC_CommandTable), 344

Exécution d'événement, 58

Exécution du programme
 Présentation, 53
 structure en blocs, 57

Exemples, commande de mouvement
 caractéristiques de vitesse du référencement
 MC, 322
 Configuration de fréquence de sorties d'impulsions pour la CPU 1217C, 278
 configuration d'une table de commande de mouvement d'un objet technologique, 308
 Configurations de fréquence de sorties d'impulsions pour les CPU 1211C, CPU 1212C, CPU 1214C et CPU 1215C, 279

Exemples, communication
 Adressage d'esclave AS-i, 175
 Communication CPU avec connexions d'émission et de réception distinctes, 158
 Communication CPU avec une connexion d'émission et de réception commune, 159
 Communication CPU par le biais de connexions TSEND_C ou TRCV_C, 160
 Protocoles de communication PROFINET, 156
 telecontrol, 186

Exemples, divers
 Ajout d'un type de données à une variable via AT, 71
 fonctions de traçage et d'analyse logique, 374
 glisser-déplacer entre les éditeurs, 35
 tranche de type de données de variable, 70

Exemples, instructions

CALCULATE, 46

Exemples, PID

PID_3Step, paramètres de configuration, 232

PID_Compact, paramètres de configuration, 231

PID_Temp, paramètres de configuration, 234

Extension des fonctionnalités du S7-1200, 18

F

FAQ, 4

FB (bloc fonctionnel), 100

FC (fonction), 100

FLOOR, 116

Fonction (FC), 100

protection du savoir-faire, 92

Fonctions mathématiques, 46, 117

Fonctions, nouvelles, 26

Forçage permanent, 359

cycle, 360

entrées et sorties, 360

mémoire I, 360

Mémoire I, 359

périphérie d'entrée, 359, 360

Fractionner éditeurs

Mise en route, 40, 44, 44

Fréquence, bits de mémento de cadence, 86

Fréquences d'entrée d'horloge HSC, 407

G

Génération de DB de page Web personnalisée, 272

Gestion des options (commande de configuration), 81

GET, 177

GET (lire les données d'une CPU distante)

configuration de la liaison, 164

GET_DIAG, 134

Get_IM_Data, 134

Glisser-déplacer entre les éditeurs, 35

H

Heure

configuration de la CPU en ligne, 365

homologation ATEX, 383

Homologation CE, 381

homologation C-Tick, 383

homologation cULus, 382

Homologation de certification coréenne, 384

homologation FM, 382

Homologation pour le domaine maritime, 384

Homologations

ATEX, 383

CE, 381

Certification coréenne, 384

C-Tick, 383

cULus, 382

FM, 382

Maritime, 384

HSC (compteur rapide)

configuration, 145, 145

fonctionnement, 135, 137

HTTP, connexions HTTP du serveur Web, 268

I

IHM

Configuration de la communication PROFINET, 147

Ecran, 50

Liaison IHM, 49

Liaison réseau, 49

Mise en route, 48, 50

Immunité aux pointes de tension, 386

Indicateur Premier cycle, 85

Information sur le programme

Dans la structure d'appel, 132

Informations pour nous contacter, 5, 79

Insertion d'instructions

Favoris, 31

Glisser-déplacer, 30

Insertion d'opérations

Glisser-déplacer entre les éditeurs, 35

Insertion d'un appareil

CPU non spécifiée, 77, 370

Installation

conseils, 23

dégagement, 25

dimensions de montage, 22

écoulement d'air, 25

modules d'entrées-sorties (SM), 19

présentation, 23

refroidissement, 25

zone thermique, 22, 25

Instruction CTRL_PWM, 126

Instructions

ajouter des entrées ou sorties aux opérations

CONT ou LOG, 31

CALCULATE, 46

CALCULER (calculer), 117

CEIL (plafonner), 116

colonnes et en-têtes, 34

Comparaison, 113

Compteur, 124

CONV (convertir), 115
Copier valeur (MOVE), 113
Copier zone (MOVE_BLK), 113
Copier zone contiguë (UMOVE_BLK), 113
CTRL_PWM, 126
état, 356
État de DEL, 133
EtatsAppareil, 134
EtatsModule, 134
Favoris, 31
FLOOR, 116
forçage permanent, 360
GET, 177
GET_DIAG, 134
Get_IM_Data, 134
Glisser-déplacer, 30
Glisser-déplacer entre les éditeurs, 35
graduation valeurs analogiques, 47
HSC (compteur rapide), 135, 137
Insertion, 30
instructions extensibles, 32
Logiques sur bits, 110
MC_ChangeDynamic (modifier les réglages dynamiques pour l'axe), 347
MC_CommandTable, 344
MC_Halt (pause de l'axe), 332
MC_Home (référencer l'axe), 329
MC_MoveAbsolute (positionnement absolu de l'axe), 334
MC_MoveJog (déplacer l'axe en mode Manuel à vue), 341
MC_MoveRelative (positionnement relatif de l'axe), 336
MC_MoveVelocity (déplacer l'axe à une vitesse prédéfinie), 338
MC_Power (libérer/bloquer l'axe), 325
MC_ReadParam (lire les paramètres de l'objet technologique), 351
MC_Reset (confirmer l'erreur), 328
MC_WriteParam (écrire dans les paramètres d'un objet technologique), 349
Mise en route, 44
Motion Control, 324
NORM_X (normaliser), 116
PID_Compact (régulateur PID universel avec fonction d'optimisation intégrée), 203
PID_Temp (régulateur PID universel qui permet de gérer la régulation de la température), 218
PUT, 177
ROUND, 115
SCALE_X (graduer), 116
TRCV_C, 154
TRUNC (tronquer), 115
TSEND_C, 154
versions des instructions, 34
visualisation, 356
visualisation de l'état ou de la valeur, 355
WWW, 273
Instructions Ethernet
 TRCV_C, 154
 TSEND_C, 154
Instructions extensibles, 32
Instructions logiques sur bits, 110
Instructions Motion Control, 324
Interface PROFINET
 Propriétés de l'adresse Ethernet, 87
Interface utilisateur
 Projet et vues de portail STEP 7, 29
Interfaces de communication
 ajouter des modules, 80
 CB 1241 RS485, 443
 CM 1241 RS232, 445
 RS232 et RS485, 193
 tableau comparatif des modules, 18
IO-Link
 Composants, 375
 configuration, 376
 données, 376
 mise sous tension, 375
 profil d'appareil, 376
 vue d'ensemble de la technologie, 375
ISO sur TCP
 configuration des liaisons, 163
 ID de liaison, 157
 mode ad hoc, 157
 paramètres, 161
ISO sur TCP, protocole, 156

J

Journal de données
 présentation générale du journal de données, 127

L

Liaison active/passive, 163
Liaison appareil local/partenaire, 163
Liaison avec une CPU, une carte mémoire ou un mot de passe, 93
Liaison dynamique, 93
Liaison réseau
 Appareils IHM, 49
 connexion d'appareils, 148

- Liaisons
 configuration, 161
 ID de liaison, 157
 liaison S7, 178
 partenaires, 163, 179
 protocoles Ethernet, 178
 Serveur Web, 268
 types de communication, 149
 types de liaisons multinoeuds, 178
- Liaisons multinoeuds
 protocoles Ethernet, 178
 types de liaisons, 178
- Ligne d'assistance, 5
- LOG (logigramme), 106
- M**
- Maître IO-Link
 exemples de raccordements, 379
 illustration, 377
 Schéma fonctionnel, 378
- Manuels, 4
- Masque de sous-réseau, 87
- MC_ChangeDynamic (modifier les réglages dynamiques pour l'axe), 347
- MC_CommandTable, 344
- MC_Halt (pause de l'axe), 332
- MC_Home (référencer l'axe), 329
- MC_MoveAbsolute (positionnement absolu de l'axe), 334
- MC_MoveJog (déplacer l'axe en mode Manuel à vue), 341
- MC_MoveRelative (positionnement relatif de l'axe), 336
- MC_MoveVelocity (déplacer l'axe à une vitesse prédéfinie), 338
- MC_Power (libérer/bloquer l'axe), 325
- MC_ReadParam (lire les paramètres de l'objet technologique), 351
- MC_Reset (confirmer l'erreur), 328
- MC_WriteParam (écrire dans les paramètres d'un objet technologique), 349
- Mémoire
 adresses de périphérie d'entrée (table de forçage permanent), 359
 Mémento de cadence, 84
 Mémento système, 84
 Mémoire de chargement, 62
 Mémoire de travail, 62
 Mémoire rémanente, 62
 Mémoire temporaire (L), 66
- Mémoire de chargement, 16, 62
 pages Web personnalisées, 271
- Mémoire de travail, 16, 62
 Mémoire globale, 66
 Mémoire I
 adresses de périphérie d'entrée (table de forçage permanent), 359
 forçage permanent, 359, 360, 360
 table de forçage permanent, 359
 table de visualisation, 355
 visualisation, 355
 visualiser CONT, 356
- Mémoire image
 état, 359
 forçage permanent, 359, 360
 visualisation de l'état ou de la valeur, 355
- Mémoire rémanente, 16, 62
- Mémoire temporaire (L), 66
- Mise à jour du firmware
 depuis STEP 7, 368, 368
- Mise à l'échelle de valeurs analogiques, 47
- Mise à niveau d'une CPU V3.0 en une V4.1, 453
- Mise en file d'attente, 58
- Mise en route
 Adressage, 44
 bloc de code, 102
 bloc de programme, 102
 Boîte d'opérations, 45
 Contacts, 42
 CPU, 39
 fractionner éditeurs, 40, 44, 44
 IHM, 48, 50
 Instructions, 44
 Liaison IHM, 49
 Liaison réseau, 49
 Nouvel API, 39
 Opération arithmétique, 45
 Programme CONT, 42, 45
 Projet, 39
 variables, 40, 44
 Variables API, 40, 44
- Modbus
 versions, 34
- Mode ad hoc, TCP et ISO sur TCP, 157
- Modification
 état de l'éditeur de programme, 356
- Modification des paramètres de STEP 7, 33
- Module commutateur compact CSM 1277, 450
- Module commutateur compact, CSM 1277, 450
- Module d'alimentation
 PM1207, 450
- Module d'alimentation PM 1207, 450
- Module de communication (CM)
 ajout d'un module CM 1243-5 (maître DP), 169

- ajout d'un module maître AS-i CM 1243-2, 173
ajouter des modules, 80
Ajouter un appareil, 78
CM 1241 RS232, 445
CM 1241 RS422/RS485, 444
configuration des appareils, 75
DEL de signalisation, 193
présentation, 19
RS232 et RS485, 193
tableau comparatif, 18
Module de communication (CM), bibliothèque USS, 195
Module d'entrées-sorties (SM)
 temps de réponse transitoire pour entrées analogiques, 419
Module d'entrées-sorties Maître IO-Link, 447
Module technologique, SM 1278 4xIO-Link Master, 447
Modules
 Communication Boards (CB), 19
 Configuration des paramètres, 82, 86
 module de communication (CM), 19
 modules d'entrées-sorties (SM), 19
 processeur de communication (CP), 19
 Signal Board (SB), 19
 tableau comparatif, 18
 zone thermique, 22, 25
Modules CANopen
 021620-B, 021630-B, 451
Modules de remplacement, 38
Modules d'entrées-sorties (SM)
 AI 4 x 13 bit SM 1231, 413
 ajouter des modules, 80
 Ajouter un appareil, 78
 configuration des appareils, 75
 Maître SM 1278 4xIO-Link, 447
 présentation, 19
 représentation des entrées analogiques (courant), 418
 représentation des entrées analogiques (tension), 417
 représentation des sorties analogiques (courant), 421
 représentation des sorties analogiques (tension), 421
 SM 1221 DI 16 24 VDC, 400
 SM 1221 DI 8 24 VDC, 400
 SM 1222 DQ 16 24 VDC, 401
 SM 1222 DQ 16 Relais, 401
 SM 1222 DQ 8 24 VDC, 401
 SM 1222 DQ 8 Relais, 401
 SM 1222 DQ 8 Relais Inverseur, 401
 SM 1223 DI 1223 DI 120/230 VAC/DQ relais, 404
 SM 1223 DI 16/DQ 16 relais, 403
 SM 1223 DI 8 / DQ 8 / DQ relais, 403
 SM 1223 DI 8/DQ 8, 403, 403
 SM 1231 AI 4 x 16 bits, 413
 SM 1231 AI 4 x 16 bits TC, 427
 SM 1231 AI 4 x RTD x 16 bits, 425
 SM 1231 AI 8 x 13 bits, 413
 SM 1231 AI 8 x 16 bits TC, 427
 SM 1231 AI 8 x RTD x 16 bits, 425
 SM 1232 AQ 2 x 14 bits, 413
 SM 1232 AQ 4 x 14 bits, 413
 SM 1234 AI 4 x 13 bits/AQ 2 x 14 bits, 413
Modules d'entrées-sorties analogiques
 SM 1231, 413
 SM 1231 RTD, 425
 SM 1231 Thermocouple, 427
 SM 1232, 413
 SM 1234, 413
Modules d'entrées-sorties TOR
 SM 1221, 400
 SM 1222, 401
 SM 1223, 403, 404
Modules retirés, 38
Montage
 conseils, 23
 dégagement, 25
 dimensions, 22
 écoulement d'air, 25
 refroidissement, 25
 zone thermique, 22, 25
MRES, panneau de commande, 33, 56, 354
My Documentation Manager, 4

N

- Niveau de protection
 bloc de code, 92
 CPU, 89
 liaison avec une CPU, une carte mémoire ou un mot de passe, 93
NORM_X (normaliser), 116
Nouveau projet
 Ajouter un appareil IHM, 48
 Ecran IHM, 50
 Liaison IHM, 49
 Liaison réseau, 49
 mise en route, 39
Nouvelles fonctions, 26
Numéros de port
 affectation aux partenaires de communication, 156

O

OB, (Bloc d'organisation)
 Objets technologiques
 Motion Control, 284
 PID, 201
 Obtenir état DEL, 133
 Octet de mémento de cadence, 86
 Octet de mémento système, 85
 OPC, configuration, 187
 Opérations
 Ajouter un paramètre, 45
 Mise en route, 45
 Opérations de comparaison, 113
 Optimisation de la vanne PID, 209
 Outils en ligne et de diagnostic
 chargement dans la CPU à l'état MARCHE, 371

P

Pages HTML personnalisées
 actualisation, 270
 conception, 269
 Pages HTML, personnalisées, 269
 emplacements de page, 272
 Pages Web
 Service, assistance et documentation STEP 7, 4
 Pages Web personnalisées, 263, 269
 activation de l'opération WWW, 273
 actualisation, 270
 configuration, 272
 contraintes de mémoire de chargement, 271
 création avec un éditeur HTML, 269
 génération de blocs de programme, 272
 programmation dans STEP 7, 273
 Pages Web standard, 264
 Pages Web standards, 263
 Pages Web STEP 7, 4
 Pages Web, personnalisées, 269
 Paramétrage, 100
 Paramètres, 33
 Paramètres de mise en route, 82
 Paramètres de sortie, 100
 Phasage, 281
 PID
 algorithme PID_3Step, 200
 algorithme PID_Compact, 200
 mise en service, 249
 PID_3STEP (Régulateur PID avec fonction d'optimisation pour vannes), 209
 PID_Compact (régulateur PID universel avec fonction d'optimisation intégrée), 203

PID_Temp (régulateur PID universel qui permet de gérer la régulation de la température), 218
 présentation, 200
 Podcasts, 4
 Portail TIA
 Ajouter un appareil, 78
 Configuration de la CPU, 82, 86
 configuration des appareils, 75
 Configuration des modules, 86
 PROFINET, 87
 Priorité
 classe de priorité, 58
 priorité dans le traitement, 58
 Processeur de communication (CP)
 ajouter des modules, 80
 présentation, 19
 tableau comparatif, 18
 PROFIBUS
 adresse, 170
 adresse, configuration, 170
 ajout d'un esclave DP, 169
 ajout d'un module CM 1243-5 (maître DP), 169
 esclave, 166
 GET, 177
 liaison réseau, 148
 liaison S7, 178
 maître, 166
 module CM 1242-5 (esclave DP), 166
 module CM 1243-5 (maître DP), 166
 nombre de liaisons de communication, 152
 PUT, 177
 PROFINET, 147
 Adresse IP, 87
 GET, 177
 ID de liaison, 157
 liaison réseau, 148
 liaison S7, 178
 mode ad hoc, 157
 nombre de liaisons de communication, 152
 présentation, 156
 PUT, 177
 Test d'un réseau, 88
 types de communication, 149
 PROFINET RT, 156
 Programmation
 ajouter des entrées ou sorties aux opérations
 CONT ou LOG, 31
 algorithme PID_3Step, 200
 algorithme PID_Compact, 200
 classe de priorité, 58
 comparaison et synchronisation des blocs de code, 364

- CONT (schéma à contacts), 105
CPU non spécifiée, 77, 370
Favoris, 31
Glisser-déplacer entre les éditeurs, 35
Insertion d'opérations, 30
instructions extensibles, 32
liaison avec une CPU, une carte mémoire ou un mot de passe, 93
Linéaire, 95
LOG (logigramme), 106
Mise en route, 44
modules retirés, 38
PID_3STEP (Régulateur PID avec fonction d'optimisation pour vannes), 209
PID_Compact (régulateur PID universel avec fonction d'optimisation intégrée), 203
PID_Temp (régulateur PID universel qui permet de gérer la régulation de la température), 218
présentation de PID, 200
SCL (langage de commande structuré), 107
SCL (Structured Control Language), 108
Structurée, 95
Programmation linéaire, 95
programmation STEP 7
 pages Web personnalisées, 273
Programmation structurée, 95, 95
Programme
 acquisition des valeurs d'un DB, 362
 appel d'un bloc, 104
 Blocs d'organisation (OB), 97
 classe de priorité, 58
 copie des blocs depuis une CPU en ligne, 363
 liaison avec une CPU, une carte mémoire ou un mot de passe, 93
 mise en route, 42
 Mise en route, 45
 modèle de réseau, 42
 Modèle de réseau, 45
 Opération arithmétique, 45
 protection par mot de passe, 92
 redéfinition des valeurs initiales d'un DB, 362
Programme utilisateur
 ajouter des entrées ou sorties aux opérations
 CONT ou LOG, 31
 Blocs d'organisation (OB), 97
 copie des blocs depuis une CPU en ligne, 363
 Favoris, 31
 Glisser-déplacer entre les éditeurs, 35
 Insertion d'instructions, 30
 instructions extensibles, 32
liaison avec une CPU, une carte mémoire ou un mot de passe, 93
protection par mot de passe, 92
Projet
 accès restreint à une CPU, 89
 ajouter un appareil IHM, 48
 comparaison et synchronisation, 364
 Ecran IHM, 50
 liaison avec une CPU, une carte mémoire ou un mot de passe, 93
 liaison IHM, 49
 Liaison réseau, 49
 mise en route, 39
 Programme, 44
 protection d'accès, 89
 protéger un bloc de code, 92
 variables, 40, 44
Propriétés de CPU, pages Web personnalisées, 272
Protection contre la copie
 liaison avec une CPU, une carte mémoire ou un mot de passe, 93
Protection contre la tension inverse, 389
Protection d'accès, CPU, 89
Protection du savoir-faire
 protection par mot de passe, 92
Protection par mot de passe
 accès à la CPU, 89
 bloc de code, 92
 CPU, 89
 liaison avec une CPU, une carte mémoire ou un mot de passe, 93
 protection contre la copie, 93
Protocole
 Communication, 191
 Communication programmable, 191
 ISO sur TCP, 156
 Modbus, 191
 PROFINET RT, 156
 TCP, 156
 UDP, 156
 USS, 191
Protocoles Ethernet, 156
 liaisons multinoeuds, 178
PTO (Sortie de trains d'impulsions), 126
 forçage permanent impossible, 360
Pupitre opérateur, 33, 56, 354
Pupitres (HMI), 20
Pupitres opérateur, 20
PUT, 177
PUT (écrire les données sur une CPU distante)
 configuration de la liaison, 164

- PWM
 Instruction CTRL_PWM, 126
- PWM (Modulation de largeur d'impulsion)
 forçage permanent impossible, 360
- R**
- Redéfinition des valeurs initiales d'un DB, 362
- Refroidissement, 25
- Remplacement de l'appareil
 Marche à suivre, 79
 Une CPU V3.0 par une CPU V4.1, 453
- Remplacement d'une CPU V3.0 par une CPU V4.1, 453
- Réseau
 Liaison réseau, 49
- Réseau, programmation CONT, 42
- Restauration des réglages d'usine, 366, 366
- ROUND, 115
- Routeur IP, 87
- RS232 et RS485, modules de communication, 193
- S**
- SCALE_X (graduer), 116
- Schémas de câblage
 CB 1241 RS 485, 443
 CPU 1214C AC/DC/Relay, 395
 CPU 1214C DC/DC/DC, 396
 SB 1221 DI 4 200 kHz, 399
 SB 1222 DQ 4 200 kHz, 399
 SB 1223 DI 2/DQ 2 200 kHz, 399
 SB 1231 AI 1 x 12 bits, 412
 SB 1231 AI 1 x 16 bits RTD, 425
 SB 1231 AI 1 x 16 bits TC, 425
 SB 1232 AQ 1 x 12 bits, 412
 SM 1221 DI 16 24 VDC, 401
 SM 1221 DI 8 24 VDC, 401
 SM 1222 DQ 16 24 VDC, 402
 SM 1222 DQ 16 Relais, 402
 SM 1223 DI 16 VDC / DQ 16 24 VDC, 403
 SM 1223 DI 16 VDC / DQ 16 relais, 403
 SM 1223 DI 8 120/230 VAC/DQ 8 relais, 404
 SM 1231 AI 8 x 13 bits, 414
 SM 1231 RTD 4 x 16 bits, 426
 SM 1231 RTD 8 x 16 bits, 426
 SM 1232 AQ 4 x 13 bits, 414
 SM 1234 AI 4 x 13 bits/AQ 2 x 14 bits, 414
 SM 1278 IO-Link maître, 449
- SCL (langage de commande structuré)
 algorithme PID_Compact, 200
- État de DEL, 133
 présentation, 107
- SCL (Structured Control Language)
 algorithme PID_3Step, 200
 arrondir, 115
 CEIL (plafonner), 116
 CONV (convertir), 115
 éditeur de programmes, 108
 EtatsAppareil, 134
 EtatsModule, 134
 FLOOR, 116
 GET_DIAG, 134
 Get_IM_Data, 134
 MC_ChangeDynamic (modifier les réglages dynamiques pour l'axe), 347
 MC_CommandTable, 344
 MC_Halt (pause de l'axe), 332
 MC_Home (référencer l'axe), 329
 MC_MoveAbsolute (positionnement absolu de l'axe), 334
 MC_MoveJog (déplacer l'axe en mode Manuel à vue), 341
 MC_MoveRelative (positionnement relatif de l'axe), 336
 MC_MoveVelocity (déplacer l'axe à une vitesse prédéfinie), 338
 MC_Power (libérer/bloquer l'axe), 325
 MC_ReadParam (lire les paramètres de l'objet technologique), 351
 MC_Reset (confirmer l'erreur), 328
 MC_WriteParam (écrire dans les paramètres d'un objet technologique), 349
 NORM_X (normaliser), 116
 PID_3STEP (Régulateur PID avec fonction d'optimisation pour vannes), 209
 PID_Compact (régulateur PID universel avec fonction d'optimisation intégrée), 203
 PID_Temp (régulateur PID universel qui permet de gérer la régulation de la température), 218
 présentation de PID, 200
 SCALE_X (graduer), 116
 section Var, 108
 tronquer, 115
- Sécurité
 CPU, 89
 liaison avec une CPU, une carte mémoire ou un mot de passe, 93
 protection contre la copie, 93
 protection d'accès, 89
 protection du savoir-faire pour un bloc de code, 92
- Séquence de mouvement (MC_CommandTable), 344

- Serveur Web, 263
 connexions HTTP maximum, 268
 contraintes, 267
- Service et assistance, 5
- Signal Board (SB)
 temps de réponse transitoire pour entrées analogiques, 419
- Signal boards (SB)
 Ajouter un appareil, 78
- Signal Boards (SB)
 ajouter des modules, 80
 configuration des appareils, 75
 présentation, 19
 représentation des entrées (courant), 418
 représentation des entrées (tension), 417
 représentation des sorties analogiques (courant), 421
 représentation des sorties analogiques (tension), 421
 SB 1221 4 DI 24 VDC 200 kHz, 397
 SB 1221 4 DI 5 VDC 200 kHz, 397
 SB 1222 4 DQ 24 VDC 200 kHz, 397
 SB 1222 4 DQ 5 VDC 200 kHz, 397
 SB 1223 2 DI/2 DQ 24 VDC, 397
 SB 1223 DI/DQ 24 VDC 200 kHz, 397
 SB 1223 DI/DQ 5 VDC 200 kHz, 397
 SB 1231 AI 1 x 12 bits, 412
 SB 1231 AI 1 x 16 bits RTD, 424
 SB 1231 AI 1 x 16 bits TC, 424
 SB 1231 RTD, 424
 SB 1231 TC, 424
 SB 1232 AQ 1 x 12 bits, 412
- Signal Boards analogiques
 SB 1231, 412
 SB 1232, 412
- Signal Boards TOR
 SB 1221, 397
 SB 1222, 397
 SB 1223, 397
- SM et SB
 tableau comparatif, 18
- SMS, 186
- Sortie de trains d'impulsions (PTO), 126
- Sorties TOR
 Spécifications pour relais, CPU 24 V CC, SM et SB, 408
- Sources d'informations, 4
- STEP 7
 acquisition des valeurs d'un DB, 362
 ajouter des entrées ou sorties aux opérations CONT ou LOG, 31
 ajouter des modules, 80
- Ajouter un appareil, 78
AS-i, 174
boutons MARCHE/ARRET, 33
classe de priorité (OB), 58
comparaison et synchronisation, 364
Configuration de la CPU, 82, 86
configuration des appareils, 75
Configuration des modules, 82, 86
Configuration HSC, 145
copie des blocs depuis une CPU en ligne, 363
entrées et sorties affichées ou masquées, 32
Favoris, 31
forçage permanent, 359, 360
Glisser-déplacer entre les éditeurs, 35
Insertion d'opérations, 30
liaison réseau, 148
mémoire tampon de diagnostic, 365
Modification des paramètres, 33
modules retirés, 38
Port Ethernet, 87
PROFIBUS, 170
PROFINET, 87
protection par mot de passe, 92
pupitre opérateur, 33, 56
Pupitre opérateur, 354
redéfinition des valeurs initiales d'un DB, 362
visualisation, 355, 356
Vue du portail et vue du projet, 29
- Structure d'appel, 132
- Structure du programme, 95
- Surveillance de l'utilisation de la mémoire, en ligne, 355
- Surveillance du temps de cycle, 354
- T**
- Table de forçage permanent
 adressage de la périphérie d'entrée, 359
 forçage permanent, 359, 360
- Table de visualisation
 forçage permanent, 130
 visualisation, 355
- Tableau comparatif
 Modèles de CPU, 16
 modules, 18
- Tableau de comparaison
 Appareils IHM, 20
- Tables de visualisation, 357
- Task Cards
 colonnes et en-têtes, 34
- TCON
 configuration, 163

- ID de liaison, 157
- paramètres de liaison, 161
- TCON_Param, 161
- TCP
 - configuration des liaisons, 163, 163
 - ID de liaison, 157
 - mode ad hoc, 157
 - paramètres, 161
 - protocole, 156
- Telecontrol, 182
- TeleService via GPRS, 182
- Temporisations
 - quantité, 17
 - taille, 17
- Temps d'attente, 58
- Tensions nominales, 388, 388
- Test du programme, 130
- TIA Portal, vue du portail et vue du projet, 29
- Trace, fonction, 374
- Traitement image
 - état, 356
 - visualisation, 356
- Tranche (d'un type de données de variable), 69
- TRCV
 - ID de liaison, 157
- TRCV (receive data via Ethernet (TCP))
 - mode ad hoc, 157
- TRCV_C
 - mode ad hoc, 157
 - paramètres de liaison, 161
- TRCV_C instruction, 154
- TRCV_C (receive data via Ethernet (TCP))
 - ID de liaison, 157
- TRCV_C (receive data via Ethernet (TCP))
 - configuration, 163
- TRUNC (tronquer), 115
- TSAP (points d'accès au service transport), 165
 - instructions pour l'affectation aux appareils, 156
- TSEND
 - ID de liaison, 157
- TSEND_C
 - paramètres de liaison, 161
- TSEND_C (send data via Ethernet (TCP))
 - configuration, 163
 - ID de liaison, 157
- TSEND_C instruction, 154
- TURCV
 - paramètres de liaison, 161
- TURCV (receive data via Ethernet (UDP))
 - configuration, 163
- TUSEND
 - paramètres, 161
- TUSEND (send data via Ethernet (UDP))
 - configuration, 163
- Types de données, 63
 - DTL, 64

U

UDP

- configuration des liaisons, 163
- paramètres, 161
- UDP, protocole, 156

V

Valeurs booléennes ou de bit, 67

Valeurs initiales

- acquisition et redéfinition des valeurs initiales d'un DB, 362

Variable

- ajout d'un type de données, 71
- forçage permanent, 360
- surveillance de l'état ou de la valeur, 355
- tranche, 69

Variables

- Mise en route, 40, 44

Variables API

- Mise en route, 40, 44

Versions des instructions, 34

Visualisation

- acquisition des valeurs d'un DB, 362
- État CONT, 356
- Etat CONT et utilisation de la table de visualisation, 355
- forçage permanent, 360
- Opération DEL, 133
- redéfinition des valeurs initiales d'un DB, 362
- table de forçage permanent, 359

Visualisation du programme, 130

Visualisation, appareils IHM, 20

Vue d'ensemble des modules RTD, 423

Vue d'ensemble des modules thermocouple, 423

Vue du portail, 29

- Ajouter un appareil, 78
- Configuration de la CPU, 82, 86
- Configuration des modules, 82, 86
- Configuration du port Ethernet, 87
- PROFINET, 87

Vue du projet, 29, 29

- Ajouter un appareil, 78
- configuration des appareils, 75
- Configuration des modules, 82, 86

Configuration des paramètres de la CPU, 82, 86
Configuration du port Ethernet, 87
PROFINET, 87

W

WWW, 273

Z

Zone thermique, 22, 25
Zones de mémoire
 accès immédiat, 66
 adressage des valeurs booléennes ou de bit, 67
 bloc de données, 66
 mémoire globale, 66
 mémoire image, 66
 mémoire temporaire, 66