

Modicon TM5

Extensomètre IoDrvTM5SEAI5G

Guide de la bibliothèque

05/2019

EIO0000003186.00

www.schneider-electric.com

Schneider
 Electric™

Le présent document comprend des descriptions générales et/ou des caractéristiques techniques des produits mentionnés. Il ne peut pas être utilisé pour définir ou déterminer l'adéquation ou la fiabilité de ces produits pour des applications utilisateur spécifiques. Il incombe à chaque utilisateur ou intégrateur de réaliser l'analyse de risques complète et appropriée, l'évaluation et le test des produits pour ce qui est de l'application à utiliser et de l'exécution de cette application. Ni la société Schneider Electric ni aucune de ses sociétés affiliées ou filiales ne peuvent être tenues pour responsables de la mauvaise utilisation des informations contenues dans le présent document. Si vous avez des suggestions, des améliorations ou des corrections à apporter à cette publication, veuillez nous en informer.

Vous acceptez de ne pas reproduire, excepté pour votre propre usage à titre non commercial, tout ou partie de ce document et sur quelque support que ce soit sans l'accord écrit de Schneider Electric. Vous acceptez également de ne pas créer de liens hypertextes vers ce document ou son contenu. Schneider Electric ne concède aucun droit ni licence pour l'utilisation personnelle et non commerciale du document ou de son contenu, sinon une licence non exclusive pour une consultation « en l'état », à vos propres risques. Tous les autres droits sont réservés.

Toutes les réglementations locales, régionales et nationales pertinentes doivent être respectées lors de l'installation et de l'utilisation de ce produit. Pour des raisons de sécurité et afin de garantir la conformité aux données système documentées, seul le fabricant est habilité à effectuer des réparations sur les composants.

Lorsque des équipements sont utilisés pour des applications présentant des exigences techniques de sécurité, suivez les instructions appropriées.

La non-utilisation du logiciel Schneider Electric ou d'un logiciel approuvé avec nos produits matériels peut entraîner des blessures, des dommages ou un fonctionnement incorrect.

Le non-respect de cette consigne peut entraîner des lésions corporelles ou des dommages matériels.

© 2019 Schneider Electric. Tous droits réservés.

Table des matières



	Consignes de sécurité	5
	A propos de ce manuel	7
Chapitre 1	Présentation générale	11
	Création du système de mesure	11
Chapitre 2	Programmation	13
2.1	Bloc fonction StrainGauge	14
	Ajout du bloc fonction StrainGaugeExt	15
	Présentation du bloc fonction StrainGaugeExt	16
2.2	Bloc fonction StrainGauge hérité	19
	Ajout du bloc fonction StrainGauge	20
	Présentation du bloc fonction StrainGauge	21
2.3	Étalonnage du système	23
	Étalonnage linéaire	24
	Création du premier point de référence	25
	Création du second point de référence	26
	Tarage du système	27
2.4	Mesure d'une valeur étalonnée	29
	Utilisation du système	29
Annexes	31
Annexe A	Types de données	33
	StainGauge_Error : codes d'erreur	34
	StrainGaugeParameter : paramètres d'étalonnage	35
Annexe B	Représentation des fonctions et blocs fonction	37
	Différences entre une fonction et un bloc fonction	38
	Utilisation d'une fonction ou d'un bloc fonction en langage IL	39
	Utilisation d'une fonction ou d'un bloc fonction en langage ST	43
Glossaire	47
Index	49

Consignes de sécurité



Informations importantes

AVIS

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



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



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

DANGER

DANGER signale un risque qui, en cas de non-respect des consignes de sécurité, **provoque** la mort ou des blessures graves.

AVERTISSEMENT

AVERTISSEMENT signale un risque qui, en cas de non-respect des consignes de sécurité, **peut provoquer** la mort ou des blessures graves.

ATTENTION

ATTENTION signale un risque qui, en cas de non-respect des consignes de sécurité, **peut provoquer** des blessures légères ou moyennement graves.

AVIS

AVIS indique des pratiques n'entraînant pas de risques corporels.

REMARQUE IMPORTANTE

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

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

A propos de ce manuel



Présentation

Objectif du document

Cette documentation décrit la configuration et les fonctions de l'extensomètre électronique en pont intégral.

Cette documentation décrit le bloc fonction et la variables de la bibliothèque IoDrvTM5SEAI5G.

Pour exploiter correctement ce guide, vous devez :

- posséder une bonne compréhension du TM5SEAI5G, notamment de sa conception, de ses fonctionnalités et de sa mise en œuvre dans les systèmes de commande ;
- maîtriser l'utilisation des langages de programmation de contrôleur CEI 61131-3 suivants :
 - langage à blocs fonction (FBD)
 - langage à contacts (LD)
 - littéral structuré (ST)
 - liste d'instructions (IL)
 - diagramme fonctionnel en séquence (SFC)

Champ d'application

Ce document a été actualisé pour le lancement d'EcoStruxure™ Machine Expert V1.1.

Document(s) à consulter

Titre de documentation	Référence
Modicon M258 Logic Controller - Guide de programmation	<i>EIO0000000402 (Eng)</i> ; <i>EIO0000000403 (Fre)</i> ; <i>EIO0000000404 (Ger)</i> ; <i>EIO0000000405 (Spa)</i> ; <i>EIO0000000406 (Ita)</i> ; <i>EIO0000000407 (Chs)</i>
Modicon LMC058 Motion Controller - Guide de programmation	<i>EIO0000000408 (Eng)</i> ; <i>EIO0000000409 (Fre)</i> ; <i>EIO0000000410 (Ger)</i> ; <i>EIO0000000411 (Spa)</i> ; <i>EIO0000000412 (Ita)</i> ; <i>EIO0000000413 (Chs)</i>
Modicon TM5 - Configuration des modules d'extension - Guide de programmation	<i>EIO0000000420 (Eng)</i> ; <i>EIO0000000421 (Fre)</i> ; <i>EIO0000000422 (Ger)</i> ; <i>EIO0000000423 (Spa)</i> ; <i>EIO0000000424 (Ita)</i> ; <i>EIO0000000425 (Chs)</i>
Modicon TM5 - Modules d'E/S analogiques - Guide de référence du matériel	<i>EIO0000003203 (Eng)</i> ; <i>EIO0000003204 (Fre)</i> ; <i>EIO0000003205 (Ger)</i> ; <i>EIO0000003206 (Spa)</i> ; <i>EIO0000003207 (Ita)</i> ; <i>EIO0000003208 (Chs)</i>

Vous pouvez télécharger ces publications et autres informations techniques depuis notre site web à l'adresse : <https://www.schneider-electric.com/en/download>

AVERTISSEMENT

PERTE DE CONTROLE

- Le concepteur d'un système de commande doit envisager les modes de défaillance possibles des chemins de commande et, pour certaines fonctions de commande critiques, prévoir un moyen d'atteindre un état sécurisé en cas de défaillance d'un chemin, et après cette défaillance. Par exemple, l'arrêt d'urgence, l'arrêt en cas de surcourse, la coupure de courant et le redémarrage sont des fonctions de contrôle cruciales.
- Des canaux de commande séparés ou redondants doivent être prévus pour les fonctions de commande critique.
- Les liaisons de communication peuvent faire partie des canaux de commande du système. Soyez particulièrement attentif aux implications des retards de transmission imprévus ou des pannes de liaison.
- Respectez toutes les réglementations de prévention des accidents ainsi que les consignes de sécurité locales.¹
- Chaque implémentation de cet équipement doit être testée individuellement et entièrement pour s'assurer du fonctionnement correct avant la mise en service.

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

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

AVERTISSEMENT

FONCTIONNEMENT IMPREVU DE L'EQUIPEMENT

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

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

Chapitre 1

Présentation générale

Création du système de mesure

Présentation

Le capteur de l'extensomètre en pont intégral fournit un signal électrique que le module TM5SEAISG convertit en une valeur numérique brute. Cette valeur est ensuite traitée par le bloc fonction StrainGaugeExt (*voir page 16*) qui fournit une valeur étalonnée.

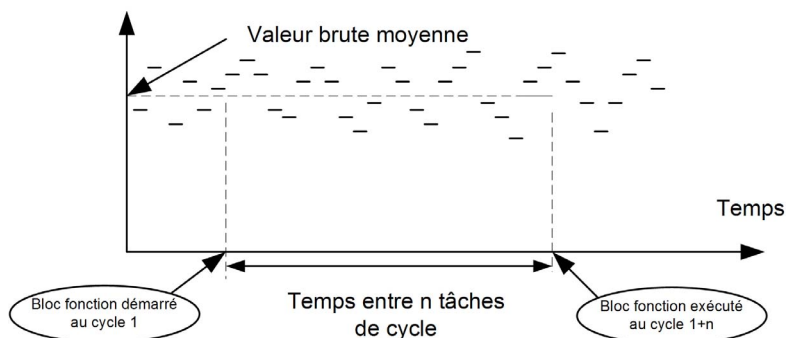
Ce bloc fonction remplit trois rôles :

- Il effectue une mesure moyenne de l'entrée TM5SEAISG pendant une période définie.
- Il définit un étalonnage linéaire correspondant à votre processus.
- Il fournit une mesure étalonnée.

NOTE : Le bloc fonction Strain Gauge n'est pas automatiquement déclaré lorsque vous ajoutez TM5SEAISG au contrôleur.

La valeur brute moyenne est calculée à partir de toutes les mesures prises par le module TM5SEAISG pendant un nombre donné de cycles de tâche. Le nombre de cycles de tâche est déterminé par l'entrée **Cycle_Number** du bloc fonction.

Valeur de AnalogInput00

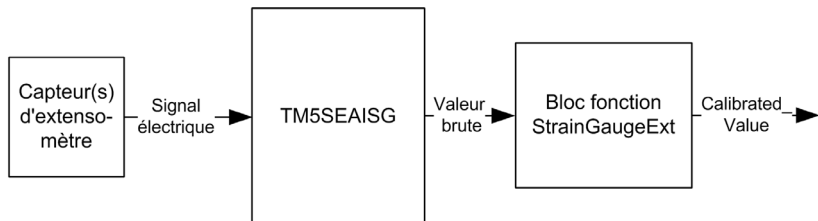


Où n correspond à la valeur de **Cycle_number**.

NOTE : la mesure fournie par le module électronique peut être faussée si les règles d'installation et de configuration de l'extensomètre en pont intégral ne sont pas suivies (*voir Modicon TM5, Modules d'E/S analogiques, Guide de référence du matériel*).

Cycle de mesure

Le schéma ci-dessous détaille le système de mesure :



Votre système de mesure est créé par la configuration (*voir Modicon TM5, Expansion Modules Configuration, Programming Guide*) de TM5SEAISG et l'utilisation du bloc fonction StrainGaugeExt.

Chapitre 2

Programmation

Présentation

Ce chapitre décrit comment utiliser le bloc fonction **StrainGaugeExt** pour étalonner votre système de mesure et obtenir une valeur étalonnée.

Contenu de ce chapitre

Ce chapitre contient les sous-chapitres suivants :

Sous-chapitre	Sujet	Page
2.1	Bloc fonction StrainGauge	14
2.2	Bloc fonction StrainGauge hérité	19
2.3	Étalonnage du système	23
2.4	Mesure d'une valeur étalonnée	29

Sous-chapitre 2.1

Bloc fonction StrainGauge

Contenu de ce sous-chapitre


Ce sous-chapitre contient les sujets suivants :

Sujet	Page
Ajout du bloc fonction StrainGaugeExt	15
Présentation du bloc fonction StrainGaugeExt	16

Ajout du bloc fonction StrainGaugeExt

Procédure

Pour ajouter et créer une instance d'un bloc fonction **StrainGaugeExt**, procédez comme suit :

Etape	Action
1	Sélectionnez l'onglet Bibliothèques dans le Catalogue de logiciels et cliquez sur Bibliothèques . Sélectionnez Contrôleur → M258 ou LMC058 → IoDrvTM5SEISG → StrainGaugeExt dans la liste, puis déplacez l'élément vers la fenêtre POU par glisser-déposer.
2	Créez l'instance du bloc fonction en cliquant sur : 
3	Les entrées et sorties sont détaillées dans la section Description des variables d'E/S (<i>voir page 17</i>).

Présentation du bloc fonction StrainGaugeExt

Présentation

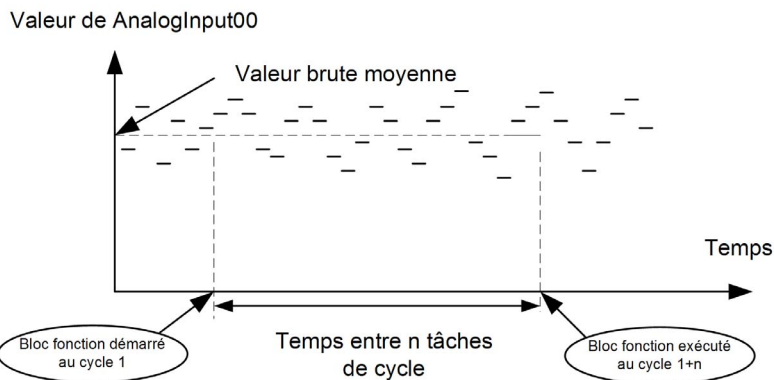
Le bloc fonction **StrainGaugeExt** est une version étendue du bloc fonction **StrainGauge**, qui permet d'effectuer une mesure de poids continue sur tout type de bus (TM5 et CANopen).

Le bloc fonction **StrainGaugeExt** s'utilise avec TM5SEASIG dans des architectures locales, distantes et distribuées.

Le bloc fonction `StrainGauge` exécute trois fonctions :

- Il mesure la valeur moyenne de l'entrée TM5SEASIG pendant une période définie.
- Il définit un étalonnage linéaire correspondant à votre processus.
- Il fournit une mesure étalonnée.

La valeur brute moyenne est calculée à partir de toutes les mesures prises par le module TM5SEASIG pendant un nombre donné de cycles de tâche. Le nombre de cycles de tâche est déterminé par l'entrée `Cycle_Number` du bloc fonction.



Où `n` correspond à la valeur de `Cycle_number`.

Présentation du bloc fonction StrainGaugeExt



Représentation en IL et en ST

Pour voir la représentation générale en langage IL ou ST, reportez-vous au chapitre Représentation des fonctions et des blocs fonctionnels (*voir page 37*).

Description des variables d'E/S

Ce tableau indique les variables d'entrée :

Entrée	Type	Initiale	Commentaire
xEnable	BOOL	–	TRUE = action en cours d'exécution. FALSE = action arrêtée ; les sorties xDone, xBusy, xError et iError sont réinitialisées.
AnalogInput	DINT	CST_INVALID_VALUE	Valeur brute donnée par le module StrainGauge A mapper via une variable avec AnalogInput00 dans : <ul style="list-style-type: none"> ● I/O Mapping du module TM5SEAISG ou ● CANopen I/O Mapping du DTM TM5/TM7 si le module StrainGauge est utilisé avec l'interface TM5 CANopen.
Tare_Enable	BOOL	FALSE	TRUE = active la fonction de tarage.
Ref1_Enable	BOOL	FALSE	TRUE = active la mesure du point de référence numéro 1.
Ref2_Enable	BOOL	FALSE	TRUE = active la mesure du point de référence numéro 2.
Cycle_number	BYTE	1	Nombre de cycles de tâche, utilisé pour calculer la moyenne de la valeur brute contenue dans AnalogInput00 (doit être différent de 0).
xContinuous	BOOL	FALSE	Mode En cours d'exécution : <ul style="list-style-type: none"> ● TRUE = mesure continue. ● FALSE = mesure unique.
s_strainGaugeParameter	StrainGaugeParameter (<i>voir page 35</i>)	–	Valeurs de tarage et d'étalonnage.

Ce tableau indique les variables de sortie :

Sortie	Type	Initiale	Commentaire
xDone	BOOL	-	TRUE = indique que l'action a abouti. L'exécution du bloc fonction est terminée.
xBusy	BOOL	-	TRUE = indique que l'exécution du bloc fonction est en cours.
xError	BOOL	-	TRUE = indique qu'une erreur a été détectée et que le bloc fonction a annulé l'action. L'exécution du bloc fonction est terminée.
xReady	BOOL	FALSE	TRUE = indique que Calibrated_value est valide.
eError	StainGauge_Error <i>(voir page 34)</i>	0	Si la valeur de xError est TRUE : type de l'erreur détectée.
Calibrated_value	DINT	CST_INVALID_VALUE	Valeur calculée après l'étalonnage du bloc fonction.

Sous-chapitre 2.2

Bloc fonction StrainGauge hérité

Contenu de ce sous-chapitre


Ce sous-chapitre contient les sujets suivants :

Sujet	Page
Ajout du bloc fonction StrainGauge	20
Présentation du bloc fonction StrainGauge	21

Ajout du bloc fonction StrainGauge

Procédure

Pour ajouter et créer l'instance d'un bloc fonction **StrainGauge**, procédez comme suit :

Etape	Action
1	Sélectionnez l'onglet Bibliothèques dans le Catalogue de logiciels et cliquez sur Bibliothèques . Sélectionnez Contrôleur → M258 ou LMC058 → IoDrvTM5SEASG → Hérité → StrainGauge dans la liste, puis déplacez l'élément vers la fenêtre POU par glisser-déposer.
2	Créez l'instance du bloc fonction en cliquant sur : 
3	Les entrées et sorties sont détaillées dans la section Description des variables d'E/S (<i>voir page 22</i>).

Présentation du bloc fonction StrainGauge

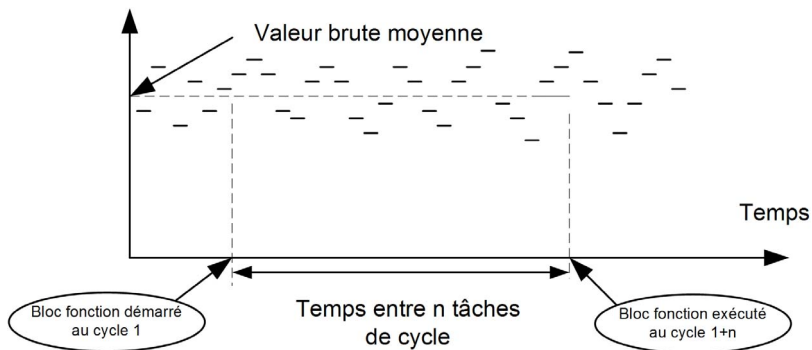
Présentation

Le bloc fonction `StrainGauge` remplit trois fonctions :

- Il effectue une mesure moyenne de l'entrée `TM5SEAISG` pendant une période définie.
- Il définit un étalonnage linéaire correspondant à votre processus.
- Il fournit une mesure étalonnée.

La valeur brute moyenne est calculée à partir de toutes les mesures prises par le module `TM5SEAISG` pendant un nombre donné de cycles de tâche. Le nombre de cycles de tâche est déterminé par l'entrée `Cycle_Number` du bloc fonction.

Valeur de `AnalogInput00`



Où `n` correspond à la valeur de `Cycle_number`.

Représentation du bloc fonction StrainGauge



Représentation en IL et en ST

Pour voir la représentation générale en langage IL ou ST, reportez-vous au chapitre Représentation des fonctions et blocs fonction (*voir page 37*).

Description des variables d'E/S

Le tableau suivant décrit les variables d'entrée :

Entrée	Type	Valeur initiale	Commentaire
xExecute	BOOL	–	Lors d'un front montant, lance l'exécution du bloc fonction. Sur le front descendant, réinitialise les sorties du bloc fonction à la fin de son exécution.
Module_Ref	TM5_STRAINGAUGE	–	Référence du module électronique d'extension TM5SEAISG.
Tare_Enable	BOOL	FALSE	TRUE = active la fonction de tarage.
Ref1_Enable	BOOL	FALSE	TRUE = active la mesure du point de référence numéro 1.
Ref2_Enable	BOOL	FALSE	TRUE = active la mesure du point de référence numéro 2.
Cycle_number	DWORD	0	Nombre de cycles de tâche, utilisé pour calculer la moyenne de la valeur brute contenue dans AnalogInput00 (doit être différent de 0).
s_strainGaugeParameter	StrainGaugeParameter <i>(voir page 35)</i>	–	Valeurs de tarage et d'étalonnage.

Le tableau suivant décrit les variables de sortie :

Sortie	Type	Valeur initiale	Commentaire
xDone	BOOL	–	TRUE = indique que l'action a abouti. L'exécution du bloc fonction est terminée.
xBusy	BOOL	–	TRUE = indique que l'exécution du bloc fonction est en cours.
xError	BOOL	–	TRUE = indique qu'une erreur a été détectée et que le bloc fonction a annulé l'action. L'exécution du bloc fonction est terminée.
eError	StainGauge_Error <i>(voir page 34)</i>	0	Lorsque xError est TRUE : type de l'erreur détectée.
Calibrated_value	DINT	FF80 0000 hex	Valeur calculée après l'étalonnage du bloc fonction <i>(voir page 23)</i> .

Sous-chapitre 2.3

Etalonnage du système

Contenu de ce sous-chapitre

Ce sous-chapitre contient les sujets suivants :

Sujet	Page
Etalonnage linéaire	24
Création du premier point de référence	25
Création du second point de référence	26
Tarage du système	27

Étalonnage linéaire

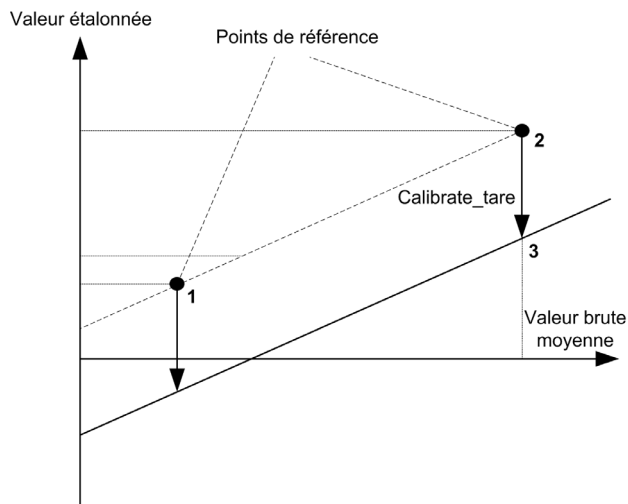
Présentation

Le bloc fonction TM5 *StrainGauge* fournit une mesure étalonnée. Vous devez étalonner votre système avant d'effectuer une mesure.

L'étalonnage du système s'effectue en trois étapes :

Etape	Action
1	Définissez un premier point de référence.
2	Définissez un second point de référence.
3	Définissez une tare.

La mesure étalonnée s'effectue par interpolation linéaire :



La droite étalonnée est enregistrée dans une variable de type `StrainGaugeParameter` (voir page 35).

NOTE : pour définir la droite d'étalonnage, il est recommandé de choisir deux points de référence : le premier à 10 ou 20 % de la valeur nominale et le second à 50 ou 60 % de la valeur nominale. Le premier à 10 ou 20 % de la valeur nominale et le second à 50 ou 60 % de la valeur nominale.

Création du premier point de référence

Conditions

Les conditions suivantes doivent être remplies :

- `Module_Ref` doit avoir une valeur correcte (*voir page 22*).
- `Cycle_Number` doit avoir une valeur correcte (*voir page 22*).

Procédure

Cette procédure définit le champ `Raw_Ref1` de la structure `s_strainGaugeParameter`.

Pour créer le second point de référence, procédez comme suit :

Etape	Action
1	Créez et stabilisez les conditions représentatives de la mesure requise pour le premier point de référence.
2	Réglez les entrées du bloc fonction <code>StrainGauge</code> comme suit : Tare_Enable = 0 Ref1_Enable = 1 Ref2_Enable = 0
3	Réglez l'entrée <code>xExecute</code> du bloc fonction sur 1.
4	Lorsque <code>xDone = 1</code> , <code>s_strainGaugeParameter.RawRef1</code> prend la valeur moyenne calculée par le bloc fonction.
5	Définissez la valeur étalonnée correspondante que vous souhaitez associer à <code>Raw_Ref1</code> dans <code>s_strainGaugeParameter.Calibrate_Ref1</code> .

Création du second point de référence

Conditions

Les conditions suivantes doivent être remplies :

- `Module_Ref` doit avoir une valeur correcte (voir page 22).
- `Cycle_Number` doit avoir une valeur correcte (voir page 22).
- La référence 1 doit être définie.

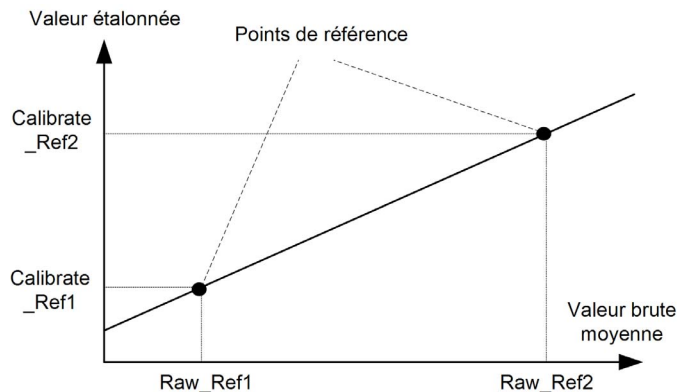
Procédure

La procédure définit le champ `Raw_Ref2` de la structure `s_strainGaugeParameter`.

Pour créer le second point de référence, procédez comme suit :

Etape	Action
1	Créez et stabilisez les conditions représentatives de la mesure requise pour le second point de référence.
2	Réglez les entrées du bloc fonction <code>StrainGauge</code> comme suit : Tare_Enable = 0 Ref1_Enable = 0 Ref2_Enable = 1
3	Réglez l'entrée <code>xExecute</code> du bloc fonction sur 1.
4	Lorsque <code>xDone = 1</code> , <code>s_strainGaugeParameter.RawRef2</code> prend la valeur moyenne calculée par le bloc fonction.
5	Définissez la valeur étalonnée correspondante que vous souhaitez associer à <code>Raw_Ref2</code> dans <code>s_strainGaugeParameter.Calibrate_Ref2</code> .

Définissez les deux points de référence pour créer la droite d'étalonnage :



Tarage du système

Conditions

Les conditions suivantes doivent être remplies :

- `Module_Ref` doit avoir une valeur correcte (*voir page 22*).
- `Cycle_Number` doit avoir une valeur correcte (*voir page 22*).
- La référence 1 doit être définie.
- La référence 2 doit être définie.

Procédure de tarage

Cette procédure vous permet de créer un offset pour définir une valeur nette en cas de présence d'une charge, ou « tare », mesurée et indiquée par le module TM5EAISG.

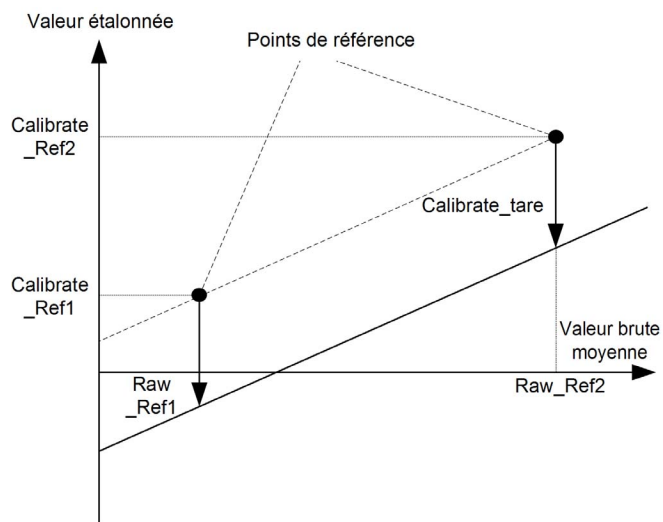
Cette procédure définit le champ `Calibrate_Tare` de la structure `s_strainGaugeParameter`.

NOTE : la tare est calculée à partir de la droite étalonnée.

Pour tarer un module électronique TM5EAISG, procédez comme suit :

Etape	Action
1	Créez et stabilisez les conditions représentatives de la mesure requise pour la tare.
2	Réglez les entrées du bloc fonction <code>StrainGauge</code> comme suit : Tare_Enable = 1 Ref1_Enable = 0 Ref2_Enable = 0
3	Réglez l'entrée <code>xExecute</code> du bloc fonction sur 1.
4	<code>s_strainGaugeParameter.Tare</code> est réglé sur la valeur étalonnée calculée par le bloc fonction.

Un offset est créé sur la droite étalonnée, précédemment définie par les deux points de référence :



Sous-chapitre 2.4

Mesure d'une valeur étalonnée

Utilisation du système

Conditions

Les conditions suivantes doivent être remplies :

- `Module_Ref` doit avoir une valeur correcte (*voir page 22*).
- `Cycle_Number` doit avoir une valeur correcte (*voir page 22*).
- La référence 1 doit être définie.
- La référence 2 doit être définie.

NOTE : si aucun paramètre d'étalonnage n'est fourni ou si ceux qui sont fournis sont incorrects, le bloc fonction renvoie une erreur (06 hex). Reportez-vous à la section Codes d'erreur (*voir page 34*).

Procédure de mesure

Après l'étalonnage du système et la définition d'une valeur de tare le cas échéant, cette procédure permet d'obtenir la valeur étalonnée mesurée par le module TM5SEASIG et calculée par le bloc fonction.

Pour mesurer une valeur, procédez comme suit :

Etape	Action
1	Définissez le bloc fonction avec la valeur d'entrée suivante : Tare_Enable = 0 Ref1_Enable = 0 Ref2_Enable = 0 xContinuous = TRUE si vous souhaitez une mesure continue (bloc fonction étendu uniquement)
2	Régalez l'entrée <code>xExecute</code> du bloc fonction sur 1.
3	Lorsque <code>xDone</code> = 1, la sortie <code>Calibrated_value</code> du bloc fonction fournit la valeur étalonnée mesurée par le module TM5SEASIG et calculée par le bloc fonction.

Annexes



Vue d'ensemble

Cette annexe reprend des extraits du guide de programmation aux fins de faciliter la compréhension technique de la documentation de la bibliothèque.

Contenu de cette annexe

Cette annexe contient les chapitres suivants :

Chapitre	Titre du chapitre	Page
A	Types de données	33
B	Représentation des fonctions et blocs fonction	37

Annexe A

Types de données

Présentation

Ce chapitre décrit les types de données de la bibliothèque HSC.

Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
StainGauge_Error : codes d'erreur	34
StrainGaugeParameter : paramètres d'étalonnage	35

StainGauge_Error : codes d'erreur**Description du type d'énumération**

Le type de données d'énumération `StainGauge_Error` contient les valeurs suivantes :

Énumérateur	Valeur	Description
<code>CALIBRATION_OK</code>	00 hex	Indique que la mesure est valide.
<code>OVERFLOW_VALUE</code>	01 hex	Indique que la valeur brute moyenne est supérieure à la valeur maximale.
<code>UNDERFLOW_VALUE</code>	02 hex	Indique que la valeur brute moyenne est inférieure à la valeur minimale.
<code>INVALID_VALUE</code>	03 hex	Indique que la valeur brute moyenne est non valide ou que le module est occupé.
<code>INVALID_CYCLETIME</code>	04 hex	Indique que l'entrée <code>Cycle_number</code> du bloc fonction a pour valeur 0.
<code>MULTIPLE_COMMAND</code>	05 hex	Indique que deux entrées du bloc fonction parmi <code>Taring_enable</code> , <code>Ref1_enable</code> et <code>Ref2_enable</code> sont réglées sur 1.
<code>INVALID_CALIBRATION_PARAMETERS</code>	06 hex	Indique que les paramètres d'étalonnage (<i>voir page 23</i>) sont non valides.
<code>INVALID_MODULE_REF</code>	07 hex	Indique que la référence du module dans l'entrée <code>Module_Ref</code> est incorrecte.

StrainGaugeParameter : paramètres d'étalonnage

Description du type énumération

Le type de données d'énumération `StrainGaugeParameter` contient les valeurs suivantes :

Enumérateur	Valeur	Description
Calibrate_Tare	DINT	Offset de la valeur étalonnée.
Calibrate_Ref1	DINT	Valeur étalonnée au point de référence 1 de la droite d'étalonnage.
Calibrate_Ref2	DINT	Valeur étalonnée au point de référence 2 de la droite d'étalonnage.
Raw_Ref1	REAL	Valeur brute moyenne au point de référence 1 de la droite d'étalonnage.
Raw_Ref2	REAL	Valeur brute moyenne au point de référence 2 de la droite d'étalonnage.

Annexe B

Représentation des fonctions et blocs fonction

Présentation

Chaque fonction peut être représentée dans les langages suivants :

- IL : (Instruction List) liste d'instructions
- ST : (Structured Text) littéral structuré
- LD : (Ladder Diagram) schéma à contacts
- FBD : Function Block Diagram (Langage à blocs fonction)
- CFC : Continuous Function Chart (Diagramme fonctionnel continu)

Ce chapitre fournit des exemples de représentations de fonctions et blocs fonction et explique comment les utiliser dans les langages IL et ST.

Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
Différences entre une fonction et un bloc fonction	38
Utilisation d'une fonction ou d'un bloc fonction en langage IL	39
Utilisation d'une fonction ou d'un bloc fonction en langage ST	43

Différences entre une fonction et un bloc fonction

Fonction

Une fonction :

- est une POU (Program Organization Unit ou unité organisationnelle de programme) qui renvoie un résultat immédiat ;
- est directement appelée par son nom (et non par une instance) ;
- ne conserve pas son état entre deux appels ;
- peut être utilisée en tant qu'opérande dans des expressions.

Exemples : opérateurs booléens (AND), calculs, conversions (BYTE_TO_INT)

Bloc fonction

Un bloc fonction :

- est une POU qui renvoie une ou plusieurs sorties ;
- doit être appelé par une instance (copie de bloc fonction avec nom et variables dédiées).
- Chaque instance conserve son état (sorties et variables internes) entre deux appels à partir d'un bloc fonction ou d'un programme.

Exemples : temporisateurs, compteurs

Dans l'exemple, `Timer_ON` est une instance du bloc fonction `TON` :

```
1  PROGRAM MyProgram_ST
2  VAR
3      Timer_ON: TON; // Function Block Instance
4      Timer_RunCd: BOOL;
5      Timer_PresetValue: TIME := T#5S;
6      Timer_Output: BOOL;
7      Timer_ElapsedTime: TIME;
8  END_VAR

1  Timer_ON(
2      IN:=Timer_RunCd,
3      PT:=Timer_PresetValue,
4      Q=>Timer_Output,
5      ET=>Timer_ElapsedTime);
```

Utilisation d'une fonction ou d'un bloc fonction en langage IL

Informations générales

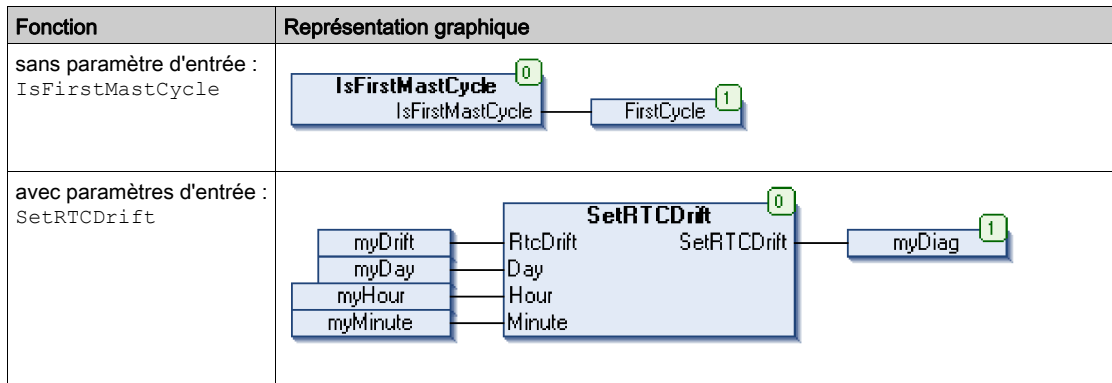
Cette partie explique comment mettre en œuvre une fonction et un bloc fonction en langage IL. Les fonctions `IsFirstMastCycle` et `SetRTCDrift`, ainsi que le bloc fonction `TON`, sont utilisés à titre d'exemple pour illustrer les mises en œuvre.

Utilisation d'une fonction en langage IL

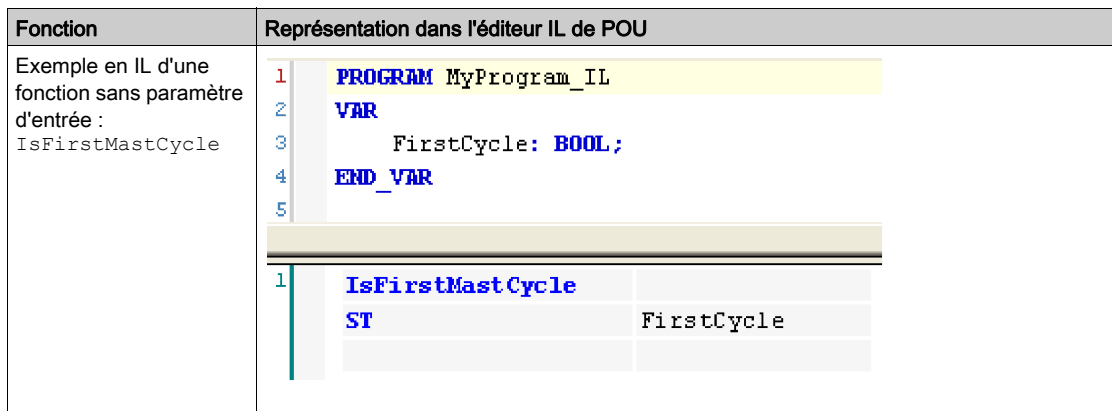
La procédure suivante explique comment insérer une fonction en langage IL :

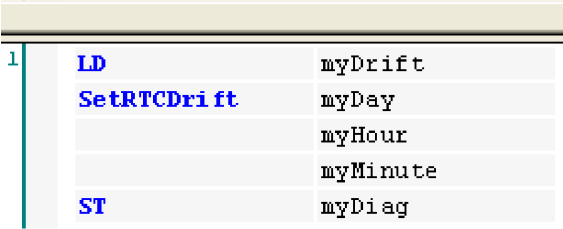
Étape	Action
1	Ouvrez ou créez une POU en langage IL (Instruction List, ou liste d'instructions). NOTE : La procédure de création d'une POU n'est pas détaillée ici. Pour plus d'informations, reportez-vous à la section Ajout et appel de POU (<i>voir EcoStruxure Machine Expert, Guide de programmation</i>).
2	Créez les variables nécessaires à la fonction.
3	Si la fonction possède une ou plusieurs entrées, chargez la première entrée en utilisant l'instruction LD.
4	Insérez une nouvelle ligne en dessous et : <ul style="list-style-type: none"> ● saisissez le nom de la fonction dans la colonne de l'opérateur (champ de gauche), ou ● utilisez l'Aide à la saisie pour sélectionner la fonction (sélectionnez Insérer l'appel de module dans le menu contextuel).
5	Si la fonction a plus d'une entrée et que l'assistant Aide à la saisie est utilisé, le nombre requis de lignes est automatiquement créé avec ??? dans les champs situés à droite. Remplacez les ??? par la valeur ou la variable appropriée compte tenu de l'ordre des entrées.
6	insérez une nouvelle ligne pour stocker le résultat de la fonction dans la variable appropriée : saisissez l'instruction ST dans la colonne de l'opérateur (champ de gauche) et le nom de la variable dans le champ de droite.

Pour illustrer la procédure, utilisons les fonctions `IsFirstMastCycle` (sans paramètre d'entrée) et `SetRTCDrift` (avec paramètres d'entrée) représentées graphiquement ci-après :



En langage IL, le nom de la fonction est utilisé directement dans la colonne de l'opérateur :



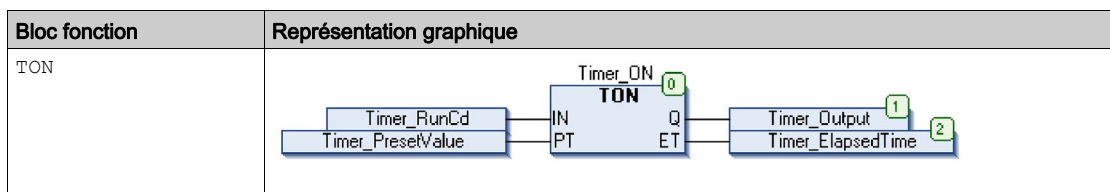
Fonction	Représentation dans l'éditeur IL de POU
Exemple IL d'une fonction avec des paramètres d'entrée : SetRTCDrift	<pre> 1 PROGRAM MyProgram_IL 2 VAR 3 myDrift: SINT (-29..29) := 5; 4 myDay: DAY_OF_WEEK := SUNDAY; 5 myHour: HOUR := 12; 6 myMinute: MINUTE; 7 myDiag: RTCSETDRIFT_ERROR; 8 END_VAR </pre> 

Utilisation d'un bloc fonction en langage IL

La procédure suivante explique comment insérer un bloc fonction en langage IL :

Étape	Action
1	Ouvrez ou créez une POU en langage IL (Instruction List, ou liste d'instructions). NOTE : La procédure de création d'une POU n'est pas détaillée ici. Pour plus d'informations, reportez-vous à la section Ajout et appel de POU (<i>voir EcoStruxure Machine Expert, Guide de programmation</i>).
2	Créez les variables nécessaires au bloc fonction (y compris le nom de l'instance).
3	L'appel de blocs fonction nécessite l'utilisation d'une instruction CAL : <ul style="list-style-type: none"> ● Utilisez l'Aide à la saisie pour sélectionner le bloc fonction (cliquez avec le bouton droit et sélectionnez Insérer l'appel de module dans le menu contextuel). ● L'instruction CAL et les E/S nécessaires sont automatiquement créées. Chaque paramètre (E/S) est une instruction : <ul style="list-style-type: none"> ● Les valeurs des entrées sont définies à l'aide de « := ». ● Les valeurs des sorties sont définies à l'aide de « => ».
4	Dans le champ CAL de droite, remplacez les ??? par le nom de l'instance.
5	Remplacez les autres ??? par une variable ou une valeur immédiate appropriée.

Pour illustrer la procédure, utilisons le bloc fonction TON représenté graphiquement ci-après :



En langage IL, le nom du bloc fonction est utilisé directement dans la colonne de l'opérateur :

Bloc fonction	Représentation dans l'éditeur IL de POU
TON	<pre> 1 PROGRAM MyProgram_IL 2 VAR 3 Timer_ON: TON; // Function Block instance declaration 4 Timer_RunCd: BOOL; 5 Timer_PresetValue: TIME := T#5S; 6 Timer_Output: BOOL; 7 Timer_ElapsedTime: TIME; 8 END_VAR 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 </pre>

Utilisation d'une fonction ou d'un bloc fonction en langage ST

Informations générales

Cette partie décrit comment mettre en œuvre une fonction ou un bloc fonction en langage ST.

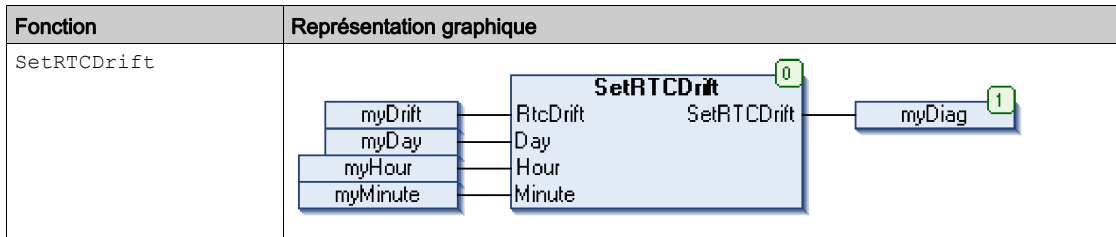
La fonction `SetRTCDrift` et le bloc fonction `TON` sont utilisés à titre d'exemple pour illustrer les mises en œuvre.

Utilisation d'une fonction en langage ST

La procédure suivante explique comment insérer une fonction en langage ST :

Étape	Action
1	Ouvrez ou créez une POU en langage ST (Structured Text ou Littéral structuré). NOTE : La procédure de création d'une POU n'est pas détaillée ici. Pour plus d'informations, reportez-vous à la section Ajout et appel de POU (<i>voir EcoStruxure Machine Expert, Guide de programmation</i>).
2	Créez les variables nécessaires à la fonction.
3	Utilisez la syntaxe générale dans l' éditeur ST de POU pour la représentation en langage ST d'une fonction. La syntaxe générale est la suivante : RésultatFonction:= NomFonction(VarEntrée1, VarEntrée2, ... VarEntréex);

Pour illustrer la procédure, utilisons la fonction `SetRTCDrift` représentée graphiquement ci-après :



La représentation en langage ST de cette fonction est la suivante :

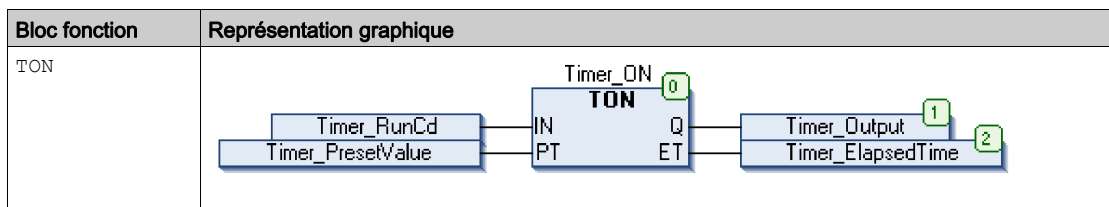
Fonction	Représentation dans l'éditeur ST de POU
SetRTCDrift	<pre>PROGRAM MyProgram_ST VAR myDrift: SINT(-29..29) := 5; myDay: DAY_OF_WEEK := SUNDAY; myHour: HOUR := 12; myMinute: MINUTE; myRTCAdjust: RTCDRIFT_ERROR; END_VAR myRTCAdjust:= SetRTCDrift(myDrift, myDay, myHour, myMinute);</pre>

Utilisation d'un bloc fonction en langage ST

La procédure suivante explique comment insérer un bloc fonction en langage ST :

Étape	Action
1	Ouvrez ou créez une POU en langage ST (Structured Text ou Littéral structuré). NOTE : La procédure de création d'une POU n'est pas détaillée ici. Pour plus d'informations sur l'ajout, la déclaration et l'appel de POU, reportez-vous à la documentation (<i>voir EcoStruxure Machine Expert, Guide de programmation</i>) associée.
2	Créez les variables d'entrée, les variables de sortie et l'instance requises pour le bloc fonction : <ul style="list-style-type: none"> • Les variables d'entrée sont les paramètres d'entrée requis par le bloc fonction. • Les variables de sortie reçoivent la valeur renvoyée par le bloc fonction.
3	Utilisez la syntaxe générale dans l' éditeur ST de POU pour la représentation en langage ST d'un bloc fonction. La syntaxe générale est la suivante : <pre>FunctionBlock_InstanceName (Input1:=VarInput1, Input2:=VarInput2, ... Ouput1=>VarOutput1, Ouput2=>VarOutput2, ...);</pre>

Pour illustrer la procédure, utilisons le bloc fonction TON représenté graphiquement ci-après :



Le tableau suivant montre plusieurs exemples d'appel de bloc fonction en langage ST :

Bloc fonction	Représentation dans l'éditeur ST de POU
TON	<pre>1 PROGRAM MyProgram_ST 2 VAR 3 Timer_ON: TON; // Function Block Instance 4 Timer_RunCd: BOOL; 5 Timer_PresetValue: TIME := T#5S; 6 Timer_Output: BOOL; 7 Timer_ElapsedTime: TIME; 8 END_VAR 1 Timer_ON(2 IN:=Timer_RunCd, 3 PT:=Timer_PresetValue, 4 Q=>Timer_Output, 5 ET=>Timer_ElapsedTime);</pre>



C

CFC

Acronyme de *continuous function chart* (diagramme fonctionnel continu). Langage de programmation graphique (extension de la norme IEC 61131-3) basé sur le langage de diagramme à blocs fonction et qui fonctionne comme un diagramme de flux. Toutefois, il n'utilise pas de réseaux et le positionnement libre des éléments graphiques est possible, ce qui permet les boucles de retour. Pour chaque bloc, les entrées se situent à gauche et les sorties à droite. Vous pouvez lier les sorties de blocs aux entrées d'autres blocs pour créer des expressions complexes.

F

FB

Acronyme de *function block*, bloc fonction. Mécanisme de programmation commode qui consolide un groupe d'instructions de programmation visant à effectuer une action spécifique et normalisée telle que le contrôle de vitesse, le contrôle d'intervalle ou le comptage. Un bloc fonction peut comprendre des données de configuration, un ensemble de paramètres de fonctionnement interne ou externe et généralement une ou plusieurs entrées et sorties de données.

I

IL

Acronyme de *instruction list*, liste d'instructions. Un programme écrit en langage IL est composé d'instructions textuelles qui sont exécutées séquentiellement par le contrôleur. Chaque instruction comprend un numéro de ligne, un code d'instruction et un opérande (voir la norme IEC 61131-3).

INT

Abréviation de *integer*, nombre entier codé sur 16 bits.

L

langage en blocs fonctionnels

Un des 5 langages de programmation de logique ou de commande pris en charge par la norme IEC 61131-3 pour les systèmes de commande. FBD est un langage de programmation orienté graphique. Il fonctionne avec une liste de réseaux où chaque réseau contient une structure graphique de zones et de lignes de connexion représentant une expression logique ou arithmétique, un appel de bloc fonction ou une instruction de retour.

LD

Acronyme de *ladder diagram*, schéma à contacts. Représentation graphique des instructions d'un programme de contrôleur, avec des symboles pour les contacts, les bobines et les blocs dans une série de réseaux exécutés séquentiellement par un contrôleur (voir IEC 61131-3).

O

octet

Type codé sur 8 bits, de 00 à FF au format hexadécimal.

P

POU

Acronyme de *program organization unit*, unité organisationnelle de programme. Déclaration de variables dans le code source et jeu d'instructions correspondant. Les POU facilitent la réutilisation modulaire de programmes logiciels, de fonctions et de blocs fonction. Une fois déclarées, les POU sont réutilisables.

S

ST

Acronyme de *structured text*, texte structuré. Langage composé d'instructions complexes et d'instructions imbriquées (boucles d'itération, exécutions conditionnelles, fonctions). Le langage ST est conforme à la norme IEC 61131-3.

T

tâche

Ensemble de sections et de sous-programmes, exécutés de façon cyclique ou périodique pour la tâche MAST, ou périodique pour la tâche FAST.

Une tâche présente un niveau de priorité et des entrées et sorties du contrôleur lui sont associées. Ces E/S sont actualisées par rapport à la tâche.

Un contrôleur peut comporter plusieurs tâches.

tare

Masse de l'emballage vide d'un produit, soustraite du poids brut pour obtenir le poids net.

V

variable

Unité de mémoire qui est adressée et modifiée par un programme.

Index



B

bibliothèque `IoDrvTM5SEAI`, 11
bloc fonction `StrainGauge` hérité, 21
blocs fonction
 `StrainGauge`, 20, 21
 `StrainGaugeExt`, 11, 15

C

cycles de tâche, définition du nombre, 11

E

étalonnage linéaire, extensomètre, 24
étalonnage, présentation de l'extensomètre, 24
extensomètre
 étalonnage linéaire, 24

F

fonctions
 différences entre une fonction et un bloc fonction, 38
 utilisation d'une fonction ou d'un bloc fonction en langage IL, 39
 utilisation d'une fonction ou d'un bloc fonction en langage ST, 43

I

installation
 système de l'extensomètre `TM5SEAI`, 11
`IoDrvTM5SEAI`, 11

M

mesure continue avec le bloc fonction `StrainGaugeExt`, 16

mesure unique avec le bloc fonction `StrainGauge`, 21
mesure, continue, 16
mesures étalonnées, `StrainGauge`, 29
mesures, `StrainGauge`, 29

P

points de référence, création pour `StrainGauge`, 25, 26

S

`StrainGauge`
 création du premier point de référence, 25
 création du second point de référence, 26
 mesure avec, 29
 tarage, 27
`StrainGauge_Error`
 types de données, 34
`StrainGauge`, bloc fonction
 ajout, 20
 entrées et sorties, 21
`StrainGaugeExt`, bloc fonction, 11
 ajout, 15
 entrées et sorties, 16
 mesures continues avec, 16
`StrainGaugeParameter`
 paramètres d'étalonnage, 24
 types de données, 35

T

tarage du système `StrainGauge`, 27
`TM5SEAI`
 création du système de mesure avec, 11
 tarage, 27
Types de données
 `StrainGauge_Error`, 34
 `StrainGaugeParameter`, 35

U

unique, mesure avec le bloc fonction Strain-Gauge, 21