

# **Dossier de formation SCE**

Siemens Automation Cooperates with Education | 05/2017

# Module 032-500 TIA Portal



Valeurs analogiques pour SIMATIC S7-1500

#### Packages SCE pour formateurs adaptés à ces dossiers de formation

#### **Automates SIMATIC**

- SIMATIC ET 200SP Open Controller CPU 1515SP PC F et HMI RT SW N° d'article: 6ES7677-2FA41-4AB1
- SIMATIC ET 200SP Distributed Controller CPU 1512SP F-1 PN Safety N° d'article: 6ES7512-1SK00-4AB2
- SIMATIC CPU 1516F PN/DP Safety N° d'article : 6ES7516-3FN00-4AB2
- SIMATIC S7 CPU 1516-3 PN/DP N° d'article: 6ES7516-3AN00-4AB3
- SIMATIC CPU 1512C PN avec logiciel et PM 1507 N° d'article : 6ES7512-1CK00-4AB1
- SIMATIC CPU 1512C PN avec logiciel, PM 1507 et CP 1542-5 (PROFIBUS) N° d'article : 6ES7512-1CK00-4AB2
- SIMATIC CPU 1512C PN avec logiciel N° d'article : 6ES7512-1CK00-4AB6
- SIMATIC CPU 1512C PN avec logiciel et CP 1542-5 (PROFIBUS) N° d'article : 6ES7512-1CK00-4AB7

#### SIMATIC STEP 7 Software for Training

- SIMATIC STEP 7 Professional V14 SP1- Licence monoposte N° d'article : 6ES7822-1AA04-4YA5
- SIMATIC STEP 7 Professional V14 SP1 Licence salle de classe 6 postes N° d'article : 6ES7822-1BA04-4YA5
- SIMATIC STEP 7 Professional V14 SP1- Licence de mise à niveau 6 postes N° d'article : 6ES7822-1AA04-4YE5
- SIMATIC STEP 7 Professional V14 SP1 Licence salle de classe 20 postes N° d'article : 6ES7822-1AC04-4YA5

Veuillez noter que les packages pour formateurs ont parfois été remplacés par de nouveaux packages.

Vous pouvez consulter les packages SCE actuellement disponibles sous : siemens.com/sce/tp

#### **Formations**

Pour les formations Siemens SCE régionales, contactez votre interlocuteur SCE régional siemens.com/sce/contact

#### Plus d'informations sur le programme SCE

siemens.com/sce

#### **Remarque d'utilisation**

Les dossiers de formation SCE pour la solution d'automatisation cohérente Totally Integrated Automation (TIA) ont été spécialement créés pour le programme "Siemens Automation Cooperates with Education (SCE)" à des fins de formation pour les instituts publics de formation et de R&D. Siemens AG n'assume aucune responsabilité quant au contenu.

Cette documentation ne peut être utilisée que pour une première formation aux produits/systèmes Siemens. Autrement dit elle peut être copiée, en partie ou en intégralité, pour être distribuée aux participants à la formation afin qu'ils puissent l'utiliser dans le cadre de leur formation. La diffusion et la duplication de cette documentation, l'exploitation et la communication de son contenu sont autorisées au sein d'instituts publics de formation et de formation continue.

Toute exception requiert au préalable l'autorisation écrite de la part de Siemens AG. Interlocuteur : Monsieur Roland Scheuerer <u>roland.scheuerer@siemens.com</u>.

Toute violation de cette règle expose son auteur au versement de dommages et intérêts. Tous droits réservés, en particulier en cas de délivrance de brevet ou d'enregistrement d'un modèle déposé.

Il est expressément interdit d'utiliser cette documentation pour des cours dispensés à des clients industriels. Tout usage de cette documentation à des fins commerciales est interdit.

Nous remercions l'Université technique de Dresde, en particulier M. le professeur Dr.-Ing. Leon Urbas et la société Michael Dziallas Engineering ainsi que toutes les personnes ayant contribué à la réalisation des dossiers de formation SCE.

# Sommaire

1	Obje	ectif	. 5
2	Con	ditions requises	. 5
3	Con	figurations matérielles et logicielles requises	. 6
4	Thé	orie	. 7
	4.1	Signaux analogiques	. 7
	4.2	Transducteurs de mesure	. 8
	4.3	Modules analogiques – Convertisseur analogique/numérique	. 8
	4.4	Types de données dans SIMATIC S7-1500	. 9
	4.5	Lecture et écriture de valeurs analogiques	10
	4.6	Normalisation des valeurs analogiques	11
5	Éno	ncé du problème	12
6	Plan	ification	12
	6.1	Commande analogique de la vitesse du convoyeur	12
	6.2	Schéma technologique	13
	6.3	Tableau d'affectation	14
7	Instr	uctions structurées par étapes	15
	7.1	Désarchiver un projet existant	15
	7.2	Création de la fonction "MOTOR_ SPEEDCONTROL"	17
	7.3	Configuration de la voie de sortie analogique	24
	7.4	Extension de la table des variables avec des signaux analogiques	25
	7.5	Appel du bloc dans le bloc d'organisation	26
	7.6	Enregistrer et compiler le programme	29
	7.7	Charger le programme	30
	7.8	Visualiser les blocs de programme	31
	7.9	Archivage du projet	33
8	Liste	e de contrôle	34
9	Exe	rcice	35
	9.1	Énoncé du problème - exercice	35
	9.2	Schéma technologique	36
	9.3	Tableau d'affectation	37
	9.4	Planification	37
	9.5	Liste de contrôle – Exercice	38
1(	) Infoi	mations complémentaires	39

# LES VALEURS ANALOGIQUES DANS SIMATIC S7-1500

# 1 Objectif

Ce chapitre présente le traitement des valeurs analogiques pour SIMATIC S7-1500 avec l'outil de programmation TIA PORTAL.

Le module décrit l'acquisition et le traitement de signaux analogiques et présente par étapes l'accès en écriture et en lecture aux valeurs analogiques dans SIMATIC S7-1500.

Les automates SIMATIC S7 énumérés au chapitre 3 peuvent être utilisés.

## 2 Conditions requises

Ce chapitre s'appuie sur le chapitre Temporisations et compteurs CEI avec SIMATIC S7 CPU1516F-3 PN/DP. Pour ce chapitre, vous pouvez par ex. utiliser le projet suivant : 032-300 IEC Timers and Counters.zap13

# 3 Configurations matérielles et logicielles requises

- 1 Station d'ingénierie : Le matériel et le système d'exploitation sont la condition de base (pour plus d'informations, voir le fichier Lisezmoi sur les DVD d'installation de TIA Portal)
- 2 Logiciel SIMATIC STEP 7 Professional dans TIA Portal à partir de V13
- 3 Automate SIMATIC S7-1500/S7-1200/S7-300, par exemple CPU 1516F-3 PN/DP à partir du firmware V1.6 avec carte mémoire et 16DI/16DO ainsi que 2AI/1AO Remarque : les entrées TOR et les entrées/sorties analogiques doivent être mises en évidence sur un pupitre.
- 4 Connexion Ethernet entre la station d'ingénierie et l'automate



3 Automate SIMATIC S7-1500

# 4 Théorie

## 4.1 Signaux analogiques

A la différence des signaux binaires qui ne peuvent prendre que les deux états "Tension disponible +24V" et "Tension indisponible 0V", les signaux analogiques sont capables (dans une certaine plage donnée) de prendre n'importe quelle valeur. Le potentiomètre est un exemple type de capteur analogique. Suivant la position de la molette, il est possible de régler une résistance quelconque jusqu'à une valeur maximum.

Exemples de grandeurs analogiques dans le technique de l'automation :

- Température -50 ... +150 °C
- Débit 0 ... 2001/min
- Vitesse -500 ... +50 tr/min
- etc.

#### 4.2 Transducteurs de mesure

A l'aide d'un transducteur de mesure, ces valeurs sont changées en tensions, en courants ou en résistances électriques. Pour l'acquisition d'une vitesse par exemple, la plage de vitesse de 500 ... 1500 tr/min peut être convertie par un transducteur de mesure en une plage de tension de 0... +10V. Pour une vitesse mesurée de 865 tr/min, le transducteur de mesure afficherait une tension de 3,65 V.



#### 4.3 Modules analogiques – Convertisseur analogique/numérique

Ces tensions, courants ou résistances électriques sont connectés à un module analogique qui numérise le signal pour poursuivre le traitement dans l'API.

Si les variables analogiques sont traitées dans un API, les valeurs de tension, de courant et de résistance lues doivent être converties en information numérique. La valeur analogique est convertie en un profil binaire. Cette conversion est appelée Conversion Analogique/Numérique (CAN). Cela signifie, par exemple, que la valeur de tension 3,65V est stockée en tant qu'information dans une série de chiffres binaires.

Pour les produits SIMATIC, le résultat de cette conversion est toujours un mot de 16 bits. Le CAN (convertisseur analogique/numérique) intégré utilisé sur le module d'entrées analogique numérise le signal analogique à acquérir et calcule sa valeur approchée par une courbe en escalier. Les paramètres les plus importants d'un CAN sont sa résolution et sa vitesse de conversion.



Plus le nombre de chiffres binaires utilisés pour la représentation numérique est grand, plus la résolution est haute. Si, par exemple, 1 seul bit était disponible pour la plage de tension 0... +10V, la seule information fournie serait que la tension mesurée est dans la fourchette de 0... +5V ou +5V... +10V. Avec 2 bits, la plage peut être divisée en quatre fourchettes individuelles, 0 ... 2,5 / 2,5 ... 5 / 5 ... 7,5 / 7,5 ... 10V. En automatique, les convertisseurs analogiques/numériques standards convertissent sur 8 ou 11 bits.

Avec un CAN 8 bits, on obtient 256 plages de valeurs, et avec un CAN 11 bits on monte à une résolution de 2048 plages.

0A/0V		20 mA/1	0 V
	11 bits		10 V : 2048 = 0,0048828 → des différences de tension <5 mV peuvent être détectées
0		2048	

### 4.4 Types de données dans SIMATIC S7-1500

Un SIMATIC S7-1500 a un grand nombre de types de données différents qui sont utilisés pour représenter les différents formats numériques. Quelques types de données de base sont listés ci-dessous.

Type de données	Taille (bit)	Plage	Exemple d'entrée constante
Bool	1	0 à 1	TRUE, FALSE, O, 1
Byte	8	16#00 à 16#FF	16#12, 16#AB
Word	16	16#0000 à 16#FFFF	16#ABCD, 16#0001
DWord	32	16#00000000 à 16#FFFFFFF	16#02468ACE
Char	8	16#00 à 16#FF	'A', 'r', '@'
Sint	8	-128 à 127	123,-123
Int	16	-32.768 à 32.767	123, -123
Dint	32	-2.147.483.648 à 2.147.483.647	123, -123
USInt	8	0 à 255	123
UInt	16	0 à 65.535	123
UDInt	32	0 à 4.294.967.295	123
Real	32	+/-1,18 x 10 -38 à +/-3,40 x 10 <sup>38</sup>	123,456, -3,4, -1,2E+12, 3,4E-3
LReal	64	+/-2,23 x 10 - <sup>308</sup> à +/-1,79 x 10 <sup>308</sup>	12345.123456789 -1.2E+40
Time	32	T#-24d_20h_31 m_23s_648ms à T#24d_20h_31 m_23s_647ms Mémorisé en tant que : - 2,147.483,648 ms à +2,147,483,647 ms	T#5m_30s 5#-2d T#1d_2h_15m_30x_45ms
Chaîne de caractères	Variable	0 à 254 caractères en taille d'octet	'ABC'

**Remarque :** Pour le traitement des valeurs analogiques, les types de données **INT** et **REAL** jouent un rôle important. En effet, les valeurs analogiques sont présentes sous forme d'entiers 16 bits au format **INT** et du fait des erreurs d'arrondi que l'on rencontre avec INT, seuls les nombres à virgule flottante de type **REAL** sont utilisables pour assurer un traitement précis.

#### 4.5 Lecture et écriture de valeurs analogiques

Les valeurs analogiques sont lues dans l'API ou écrites comme informations de mot. L'accès à ces mots s'effectue par ex. avec les opérandes :

%EW 64	mot d'entrée analogique 64
%AW 64	mot de sortie analogique 64

À chaque valeur analogique (voie) est assigné un mot d'entrée ou un mot de sortie. Le format est **,Int**` (pour integer), un entier.

L'adressage des valeurs d'entrée et des valeurs de sortie dépend de l'adressage dans la vue des appareils. Par exemple :



L'adresse de la première entrée analogique serait dans ce cas %EW 64, celle de la deuxième entrée analogique %EW 66, celle de la troisième entrée analogique %EW 68, celle de la quatrième entrée analogique EW70, celle de la cinquième entrée analogique EW72, celle de la sixième entrée analogique EW74, celle de la septième entrée analogique EW76 et celle de la huitième entrée analogique EW78.

L'adresse de la première sortie analogique serait dans ce cas %AW 64, celle de la deuxième sortie analogique %AW 66, celle de la sortie analogique %AW68 et celle de la quatrième sortie analogique AW70.

La transformation d'une valeur analogique en vue d'un traitement dans un API est la même pour les entrées et les sorties analogiques.

Les plages de valeurs numérisées ressemblent à ceci :



Ces valeurs numérisées doivent souvent être encore normalisées par un traitement dans l'API.

## 4.6 Normalisation des valeurs analogiques

Si une valeur d'entrée analogique est présente en tant que valeur numérisée dans une plage +/- 27648, elle doit habituellement être normalisée pour que les valeurs numériques correspondent à des valeurs physiques du processus.

La sortie analogique est également réalisée par la transmission d'une valeur normée qui doit être ensuite être mise à l'échelle à la valeur de la sortie +/- 27648.

Dans le TIA Portal, on aura recours à des blocs standards ou à des opérations de calculs programmés pour la normalisation et la mise à l'échelle.

Pour que cela soit aussi exact que possible, les valeurs à normer doivent être converties en format de données REAL, pour minimiser les erreurs d'arrondi.

# 5 Énoncé du problème

Ce chapitre vise à étendre le programme du chapitre Temporisations et compteurs CEI "SCE\_FR\_032-300 IEC Timers and Counters" par une fonction de commande analogique de la vitesse du convoyeur.

## 6 Planification

La programmation de la commande analogique de la vitesse du convoyeur s'effectue dans la fonction "MOTOR\_ SPEEDCONTROL" [FC10] en tant qu'extension du projet "SCE\_FR\_032-300 IEC Timers and Counters". Ce projet doit être désarchivé pour que cette fonction puisse être ajoutée. La fonction "MOTOR\_ SPEEDCONTROL" [FC10] est appelée et connectée dans le bloc d'organisation "Main" [OB1]. La commande du moteur du convoyeur doit être modifiée à –Q3 (moteur du convoyeur -M1 vitesse variable).

#### 6.1 Commande analogique de la vitesse du convoyeur

La consigne de vitesse doit être transmise à une entrée de la fonction "MOTOR\_ SPEEDCONTROL" [FC10] en tours par minute (plage : +/- 50 tr/min). Le type de données est ici virgule flottante 32 bits (Real).

Dans la fonction, il faut d'abord vérifier que la consigne de vitesse est correctement saisie dans la plage +/- 50 tr/min.

Si la consigne de vitesse n'est pas comprise dans la plage de +/- 50 tr/min, il faut attribuer à la sortie la valeur de réglage de vitesse 0 avec le type de données entier 16 bits (Int). La valeur TRUE (1) est attribuée à la valeur de retour de la fonction (Ret\_Val).

Si la consigne de vitesse est comprise dans la plage +/- 50 tr/min, cette valeur doit d'abord être normée à la plage 0...1 puis être mise à l'échelle en tant que valeur de réglage de vitesse sur la sortie analogique à +/- 27648 avec le type de donnée entier 16 bits (Int).

La sortie est connectée avec le signal -U1 (valeur de réglage de la vitesse du moteur dans les deux directions +/-10V correspondent à +/- 50 tr/min).

## 6.2 Schéma technologique

Vous voyez ici le schéma technologique de l'énoncé du problème.



Figure 1 : Schéma technologique

Schalter der Sortieranlage	Automatikbetrieb	Handbetrieb / Manual mode
Switches of sorting station	Automatic mode	-S3 Tippbetrieb -M1 vorwärts/
-P1 ein/on	-P5 gestartel/started	Manual -M1 forwards
-Q0 Hauptschalter/Main switch     -P4 aktiviert/active     -P4 aktivert/active     -P4 aktiviert/active     -P4 aktivert/active     -P4 aktivert/activert/active     -P4 aktivert	-S1 Start/start -S2 Stopp/stop	-S4 Tippbetrieb -M1 rückwärts/ Manual -M1 backwards -P7 ausgefahrenlextended -S6 Zylinder -M4 ausfahren/ cylinder -M4 extend -S5 Zylinder -M4 einfahren/ cylinder -M4 retract

Figure 2 : Pupitre de commande

## 6.3 Tableau d'affectation

DI	Туре	Désignation	Fonction	NF/NO
E 0.0	BOOL	-A1	Alarme Arrêt d'urgence ok	NF
E 0.1	BOOL	-K0	Installation "Marche"	NO
E 0.2	BOOL	-S0	Commutateur mode manuel (0) / automatique (1)	Manuel = 0 Auto=1
E 0.3	BOOL	-S1	Bouton poussoir démarrage automatique	NO
E 0.4	BOOL	-S2	Bouton poussoir arrêt automatique	NF
E 0.5	BOOL	-B1	Capteur vérin -M4 rentré	NO
E 1.0	BOOL	-B4	Capteur toboggan affecté	NO
E 1.3	BOOL	-B7	Capteur de pièce en fin de convoyeur	NO

Les signaux suivants seront nécessaires pour cette tâche, au titre d'opérandes globaux.

DO	Туре	Désignation	Fonction	
A 0.2	BOOL	-Q3	Moteur du convoyeur -M1 vitesse variable	
AW 64	BOOL	-U1	Valeur de réglage de la vitesse du moteur dans les deux directions +/-10V correspondent à +/- 50 tr/min	

#### Légende de la liste d'affectation

- DI Entrée TOR DQ Sortie TOR
- AI Entrée analogique AQ Sortie analogique

Sortie

- Entrée Q
- NF Normalement fermé (contact à ouverture)

Т

NO Normalement ouvert (contact à fermeture)

## 7 Instructions structurées par étapes

Vous trouverez ci-après des instructions pour réaliser la planification. Si vous êtes déjà expérimenté, les étapes numérotées vous suffisent. Dans le cas contraire, orientez-vous à l'aide des étapes suivantes des instructions.

#### 7.1 Désarchiver un projet existant

→ Avant de pouvoir étendre le projet "032-300 IEC Timers and Counters.zap13" du chapitre Temporisations et compteurs CEI "SCE\_FR\_032-300 IEC Timers and Counters\_S7-1500", il faut le désarchiver. Pour désarchiver un projet existant, vous devez rechercher l'archive à partir de la vue de projet sous → Projet → Désarchiver. Confirmez votre choix avec "Ouvrir".

ject	Edit	View	Insert	0			
New							
Open.			Ctrl+O	h			
Migrat	e proj	ect		8			
Close			Ctrl+W				
Save			Ctrl+S				
Save a	as	Ct	rl+Shift+S	ł			
Delete	e proje	ct	Ctrl+E				
Archive							
Retrieve							
Card R	eader	USB m	emory )				
Memo	ry card	l file	,	·			
Upgra	de						
Exit							
	ject New Open. Migrat Close Save Save Save Save Archiv Retrie Card R Memo Upgra Exit	ject Edit New Open Migrate proj Close Save Save as Delete proje Archive Retrieve Card Reader Memory card Upgrade Exit	ject Edit View New Open Migrate project Close Save Save as Ct Delete project Archive Retrieve Card Reader/USB m Memory card file Upgrade Exit	ject Edit View Insert New Open Ctrl+O Migrate project Close Ctrl+W Save Ctrl+S Save as Ctrl+Shift+S Delete project Ctrl+E Archive Retrieve Card Reader/USB memory Memory card file			

 $(\rightarrow \text{Projet} \rightarrow \text{Désarchiver} \rightarrow \text{Sélection d'une archive .zap} \rightarrow \text{Ouvrir})$ 

→ Sélectionner ensuite le répertoire cible pour enregistrer le projet désarchivé. Confirmez votre sélection par "OK".

 $(\rightarrow \text{Répertoire cible} \rightarrow \text{OK})$ 

 $\rightarrow$  Enregistrer le projet sous 032-500\_Analog\_Values\_S7-1500.

 $(\rightarrow \text{Projet} \rightarrow \text{Enregistrer sous} \dots \rightarrow 032\text{-}500\text{-}Analog\text{-}Values} \rightarrow \text{Enregistrer})$ 



## 7.2 Création de la fonction "MOTOR\_ SPEEDCONTROL"

→ Choisissez le dossier "Blocs de programme" de votre CPU 1516F-3 PN/DP et cliquez ensuite sur "Ajouter nouveau bloc" afin de créer une nouvelle fonction à cet endroit.

M Siemens - G:\Automation\032-500\_Analog\_Values\03 Project Edit View Insert Online Options Tools 📑 🔁 🖬 Save project 🔳 🐰 🏥 🗎 🗙 🏷 ± 🥵 ± Project tree Devices 00 032-500\_Analog\_Values ~ Add new device h Devices & networks CPU1516F [CPU 1516F-3 PN/DP] Device configuration & Online & diagnostics 🕶 🙀 Program blocks Add new,block Main [OB19] MOTOR\_AUTO [FB1] MOTOR\_AUTO\_DB [DB1] Technology objects External source files 📮 PLC tags C PLC data types

 $(\rightarrow \text{CPU}_1516\text{F} \text{ [CPU 1516F-3 PN/DP]} \rightarrow \text{Ajouter nouveau bloc})$ 

→ Dans le dialogue suivant, sélectionnez et nommez votre nouveau bloc avec le nom : "MOTOR\_ SPEEDCONTROL". Choisissez le langage LOG et attribuez manuellement le numéro 10. Activez la case à cocher "Ajouter nouveau et ouvrir". Cliquez sur "OK".

(→  $\rightarrow$  Nom : MOTOR\_ SPEEDCONTROL → Langue : LOG→ Numéro : 10 manuel →  $\blacksquare$  Ajouter nouveau et ouvrir → OK)

NOTOR_SPEEDCON	ITROL				
Organization block	Language: Number:	FBD 10 Manual Automatic	•		
Function block	Description: Functions are c	ode blocks or subrou	tines without dedica	ted memory.	
Function					
Data block	More				

→ Créez les variables locales indiquées ici avec les commentaires et modifiez le type de données de la variable "Return" de "Void" à "Bool".

 $(\rightarrow Bool)$ 

	Val	ues	► CF	VU151	16F [C	:PU 1	516F-3 PN/DP]	Program	blocks	MOTOR_	SPEEDCONTROL [F	C10] 💶 🖬 🖬 🗙
юł	kǎ kǎ 🦸 🔹 🍉 🗮 🚍 💬 웹 ± 월 ± 📃 😥 🥙 😡 🚳 🕸 🖳 🔛 🔛											
	MOTOR_SPEEDCONTROL											
		Na	me					Data type		Default value	Comment	
1	-00	٠	Input									
2	-00	•	Set	point_	speed			Real				
3	-00	٠	Output	t								
4		•	Ma	nipula	ted_va	riable_	_speed_AO	Int				
5		•	InOut									
6		۰.	<ad< td=""><td>ld new</td><td>/&gt;</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></ad<>	ld new	/>							
7	-00	•	Temp									
8	-00	•	Set	point_	speed	_ок		Bool				
9	-00	•	Ma	nipula	ted_va	riable_	_speed_Norm	Real				
10	-00	•	Consta	nt								
11		•	<ac< td=""><td>ld new</td><td>/&gt;</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></ac<>	ld new	/>							
12	-00	•	Return									
13	-	•	MO	TOR_S	PEEDC	ONTRO	L	Bool				
	<											>
	-		_					•				
8	1	> = 1		-	-01	-	-1=1					
-	Blo	ck	title:	Speed	contro	ol via a	nalog output					^
C	omi	mer	nt									
	E											≡
•	Network 1:											
	C	om	ment									
												~
										100	%	] <u></u>

Remarque : Veillez à utiliser le bon type de données.

→ Insérez dans le premier réseau une affectation <sup>-</sup>f=1 et avant elle un ET <sup>\*</sup>. Faites glisser ensuite le "Comparateur" "Inférieur ou égal" des "Instructions de base" sur la première entrée de l'opération ET <sup>\*</sup>.

```
(\rightarrow -[=] \rightarrow a \rightarrow Instructions de base \rightarrow Comparateur \rightarrow CMP<=)
```

					Options	_	-
a د 👻 🔹 🖦 🖿 🚍 🗩 😫 د	1± 🖃 😥 🍋 🖕 🤆			4	ių ii		
MOTOR_SPEEDCONTROL					> Favorites	_	
Name	Data type	Default value Com	ment		✓ Basic instructions		
🕣 🔻 Input				^	Name	Versi	
Setpoint_speed	Real				General		
📶 🔻 Output				- 11	Bit logic operations		h
Manipulated_variable_speed_AO	Int				Imer operations		1
InOut InOut				*	Fil Counter operations		1
<				>	Comparator operations		l
>=1 1221 -1 -011=1					E CMP ==		1
					E CMP ↔		l
Block title: Speed control via analog output				^	CMP >=		
omment					E CMP <=		l
Sala la la constanta					E CMP >		
Network 1: check setpoint speed for corre	ect input range +/- 50 r/r	nın			E CMP <		
Comment					IN_Range		
					OUT_Range		
&	?.?			=	🗉 - ОК -		
?.? -	=				-INOT_OK -		
?.? -111+	—				Variant		
					Math functions		

→ Faites glisser le "Comparateur", "Supérieur ou égal" sur la deuxième entrée de l'opération ET <sup>a</sup>

 $(\rightarrow$  Instructions de base  $\rightarrow$  Comparateur  $\rightarrow$  CMP>=)

U1516F [CPU 1516F-3 PN/DP] > Program I	olocks • MOTOR_	SPEEDCONTROI	L [FC10]	- •	×	Instructions	- III 🕨 🕨
						Options	
ta ta 🔹 🔹 🍡 🗮 🚍 💭 🕾 ta 🖏	😑 🗊 🍋 💊 🖉	- 😪 🤣 🖳 🍾			4	titi jiti	
MOTOR_SPEEDCONTROL						> Favorites	
Name	Data type	Default value	Comment			✓ Basic instructions	
1 🕘 🔻 Input					~	Name	Versi
2 📲 Setpoint_speed	Real				≡	Ceneral	
3 🕣 🔻 Output						Bit logic operations	
4 📹 🔹 Manipulated_variable_speed_AO	Int					O Timer operations	
5 📶 🔻 InOut					¥	+1 Counter operations	
<					>	<ul> <li>Comparator operations</li> </ul>	
						E CMP ==	=
<ul> <li>Block title: Speed control via analog output</li> </ul>					^	CMP >=	
Comment						E CMP <=	
					_	E CMP >	
Network 1: check setpoint speed for correct	t input range +/- 50 r/m	in				E CMP <	
Comment						IN_Range	
						OUT_Range	
<=					=	🗉 - ОК	
222						INOT_OK -	
?? — IN1	&					🕨 🛅 Variant	
?? — IN2		Math functions					
		Move operations					
?.? - *		-				🕨 🄄 Conversion operations	
						🕨 🖬 Program control operati	~
						<	>

→ Connectez maintenant les contacts dans le réseau 1 avec les constantes et les variables locales, comme indiqué ici. Les types de données dans les comparateurs seront automatiquement modifiés en "Real".



→ Dans le réseau 2, faites maintenant glisser le "Convertisseur" "NORM\_X" pour normaliser la consigne de vitesse de +/-50 tr/min à +/- 1.

 $(\rightarrow$  Instructions de base  $\rightarrow$  Convertisseur  $\rightarrow$  NORM\_X)



→ Connectez maintenant les contacts dans le réseau 2 avec les constantes et les variables locales, comme indiqué ici. Les types de données dans "NORM\_X" sont automatiquement modifiés en "Real".



→ Dans le réseau 3, faites glisser le "Convertisseur" "SCALE\_X" pour dénormaliser la consigne de vitesse de +/- 1 à la plage pour la sortie analogique +/-27648.
 (→ Instructions de base → Convertisseur → SCALE\_X)



→ Connectez ensuite les contacts dans le réseau 3 avec les constantes et les variables locales, comme indiqué ici. Les types de données dans "SCALE\_X" seront automatiquement modifiés en "Real" ou "Int".



→ Insérez une affectation <sup>-</sup>f=1 dans le quatrième réseau. Faites ensuite glisser l'ordre "Move" à partir du dossier "Transfert" dans les "Instructions de base" et déposez-le devant l'affectation.

'U1516F [CPU 1516F-3 PN/DP] > Program	blocks  MOTOR_S	PEEDCONTROL [	[FC10] 📃 🗖	∎×	Instructions	
					Options	
න් ත් 学 👻 🍉 🖿 🖬 🔛 😫 😫	± 🖃 😥 🥙 🖕 🛤	영문 😍 🧤 🛬	e 📽 🕼		liù jùi	
MOTOR_SPEEDCONTROL					> Favorites	
Name	Data type	Default value C	lomment		✓ Basic instructions	
1 📶 🔻 Input				~	Name	Versi
2 📶 = Setpoint_speed	Real			≡	🕨 🛅 General	1
3 🕣 🔻 Output				_	Bit logic operations	
4 an Manipulated_variable_speed_AO	Int 🔳				Timer operations	
5 📶 🔻 InOut				~	Figure Counter operations	
<				>	Comparator operations	-
					Hath functions	
					Move operations	
#Manipulated_	#Manipula	ted_		^	MOVE	
variable_speed_	variable_sp	peed_			Deserialize	<u>V1.1</u>
VALUE	OUT - AU				E Serialize	<u>V1.1</u>
27648 — MAX	ENO -				MOVE_BLK	
					MOVE_BLK_VARIANT	<u>V1.2</u>
<ul> <li>Network 4: Speed setpoint out of range +/-</li> </ul>	- 50 r/min -> Manipulated	variable speed AG	) = 0 / Return = TRU	E	UMOVE_BLK	
Comment				- 1		
Comment				- 1		
				-		
24				_	Variant	
				=	variant	

 $(\rightarrow$   $\neg$  Instructions de base  $\rightarrow$  Transfert  $\rightarrow$  MOVE)

→ Connectez ensuite les contacts dans le réseau 4 avec les constantes et les variables locales, comme indiqué ici. Si la consigne de vitesse ne se situe pas dans la plage +/- 50 tr/min, la valeur "0" est écrite sur la sortie analogique et la valeur TRUE est affectée à la valeur de retour (Return) de la fonction "MOTOR\_ SPEEDCONTROL".

'U1516F [CPU 1516F-3 PN/DP] • Program	blocks > MOTOR	_SPEEDCONTRO	DL [FC10]	_ @ =>	( li	nstructions	<b>.</b>		
					0	ptions			
ාස් ශ් 🖻 👻 🎭 🖿 🚍 🚍 💬 🗐 ± 😂	± 😑 🗊 🥙 😡	ا 😍 📾 😸 ا	× 00 1			tini tini			
MOTOR_SPEEDCONTROL					>	Favorites			
Name	Data type	Default value	Comment		- v	Basic instructions			
7 🔄 🔻 Temp					N N	ame	Versi		
8 📶 = Setpoint_speed_OK	Bool	1				General		~	
9 🕣 = Manipulated_variable_speed_Norm	Real					Bit logic operations		-	
10 🕣 🔻 Constant						Timer operations			
11 Add new>						+1 Counter operations			
<				>	Ъ	Comparator operations			
					= ,	1 Math functions		≡	
& >=1 ??? -I -oI[=]					<b> </b> -	Move operations			
						I MOVE			
<ul> <li>Network 4: Speed setpoint out of range +/-</li> </ul>	50 r/min -> Manipulat	ed_variable_speed	d_AO = 0 / Retur	n = TRUE	ì	Deserialize	V1.1		
Comment				_	ι.	Serialize	V1.1		
				_	ι.	MOVE BLK			
MOVE					ι.	MOVE BLK VARIANT	V1.2		
#Manie	#N	IOTOR			L .				
#Setpoint variable	e speed SPEED	CONTROL			L .				
speed_OK 🗕 EN 🛶 OUTI 🗕 AO	speed_OK - EN - OUTI - AO =								
0 - IN ENO		-				FT SWAP			
						Array DB			

→ N'oubliez pas de cliquer sur Save project. La fonction "MOTOR\_ SPEEDCONTROL" [FC10] programmée est représentée ci-après en langage LOG.



## 7.3 Configuration de la voie de sortie analogique

 $\rightarrow$  Ouvrez la "Configuration des l'appareils" avec un double-clic.



→ Vérifiez le réglage de l'adresse et la configuration de la voie de sortie analogique 0.
 (→ Adresse -A : 64...71 → Propriétés → Généralités → Sortie 0 - 3 → Sorties → Voie 0
 → Type de sortie : Tension → Plage de sortie : +/- 10 V → Réaction à l'arrêt de la CPU : Coupure)



### 7.4 Extension de la table des variables avec des signaux analogiques

→ Ouvrez la table des variables de l'installation de tri "Tag table\_sorting station" avec un double-clic.



→ Complétez la table des variables de l'installation de tri "Tag table\_sorting station" avec les variables globales pour le traitement des valeurs analogiques. Vous pouvez ajouter une entrée analogique –B8 et une sortie analogique –U1.

A_0(	nalog_Values ► CPU15161	F [CPU 1516F-3	PN/DP] ▶ PLC t		-		
			-	ays	lag tab	le_sortir	ng station [30]
							🕣 Tags 🗉 User constants
	° 🕆 🗓						
tab	le_sorting station						
Na	me	Data type	Address	Retain	Visibl	Acces	Comment
01	-\$5	Bool	%I1.6				pushbutton manual mode cylinder -M4 retract (no)
01	-56	Bool	%11.7		<b></b>	<b></b>	pushbutton manual mode cylinder -M4 extend (no)
01	-Q1	Bool	%Q0.0			<b></b>	conveyor motor -M1 forwards fixed speed
01	-Q2	Bool	%Q0.1		<b></b>	<b></b>	conveyor motor -M1 backwards fixed speed
01	-Q3	Bool	%Q0.2			<b></b>	conveyor motor -M1 variable speed
01	-M2	Bool	%Q0.3		<b></b>	<b></b>	cylinder -M4 retract
01	-MB	Bool	%Q0.4			<b></b>	cylinder -M4 extend
01	-P1	Bool	%Q0.5			<b></b>	display "main switch on"
01	-P2	Bool	%Q0.6		<b></b>	<b></b>	display "manual mode"
01	-P3	Bool	%Q0.7		<b></b>	<b></b>	display "automatic mode"
01	-P4	Bool	%Q1.0		<b></b>	<b>~</b>	display "emergency stop activated"
01	-P5	Bool	%Q1.1		<b></b>	<b>~</b>	display "automatic mode started"
01	-P6	Bool	%Q1.2		<b></b>	<b></b>	display cylinder -M4 "retracted"
01	-P7	Bool	%Q1.3		<b></b>	<b></b>	display cylinder -M4 "extended"
01	-U1	Int	%QW64			<b></b>	manipulated value speed in 2 directions +/- 10V
01	-88	Int	%IW64			<b></b>	sensor actual value speed 010V
	<add new=""></add>				<b>V</b>	<b>V</b>	
	y tab Na U U U U U U U U U U U U U U U U U U	PC         X           Name	Name         Data type           -S5         Bool           -S5         Bool           -Q1         Bool           -Q2         Bool           -Q2         Bool           -Q3         Bool           -H8         Bool           -P1         Bool           -P2         Bool           -P1         Bool           -P2         Bool           -P3         Bool           -P5         Bool           -P5         Bool           -P7         Bool	Name         Data type         Address           -S5         Bool         %I1.6           -S5         Bool         %I1.7           -Q1         Bool         %Q0.0           -Q2         Bool         %Q0.1           -Q3         Bool         %Q0.4           -Q3         Bool         %Q0.4           -P1         Bool         %Q0.4           P1         Bool         %Q0.6           -P2         Bool         %Q0.6           -P3         Bool         %Q0.7           -P4         Bool         %Q1.0           -P5         Bool         %Q1.2           -P6         Bool         %Q1.2           -P7         Bool         %Q1.2           -P7         Bool         %Q1.2           -P7         Bool         %Q1.2           -P7         Bool         %Q1.3           -U1         Int         %QW644           -B8         Int<	Name         Data type         Address         Retain           -55         Bool         %I1.6	Image: Station       Data type       Address       Retain       Visibl         S-55       Bool       %11.6       Image: Station         S-56       Bool       %11.7       Image: Station         -Q1       Bool       %Q0.0       Image: Station         -Q2       Bool       %Q0.1       Image: Station         -Q2       Bool       %Q0.1       Image: Station         -Q3       Bool       %Q0.3       Image: Station         -Q4       Bool       %Q0.3       Image: Station         -Q3       Bool       %Q0.4       Image: Station         -Q4       Bool       %Q0.4       Image: Station         -P1       Bool       %Q0.4       Image: Station         -P1       Bool       %Q0.4       Image: Station         -P1       Bool       %Q0.7       Image: Station         -P2       Bool       %Q0.7       Image: Station         Image: P3       Bool       %Q0.7       Image: Station         Image: P4       Bool       %Q1.0       Image: Station         Image: P4       Bool       %Q1.1       Image: Station         Image: P4       Bool       %Q1.3       Image: Station	Name         Data type         Address         Retain         Visibl         Acces           -55         Bool         %11.6               -56         Bool         %11.7               -Q1         Bool         %Q0.0                -Q2         Bool         %Q0.1 </td

 $(\rightarrow -U1 \rightarrow \%AW64 \rightarrow -B8 \rightarrow \%EW64)$ 

## 7.5 Appel du bloc dans le bloc d'organisation

 $\rightarrow$  Ouvrez le bloc d'organisation "Main [OB1]" par un double clic.



→ Complétez les variables locales de l'OB1 avec la variable temporaire

"Motor\_speed\_monitoring\_Ret\_Val". Celle-ci sera nécessaire pour connecter la valeur de retour de la fonction "MOTOR\_ SPEEDCONTROL".

 $(\rightarrow \text{Temp} \rightarrow \text{Motor\_speed\_monitoring\_Ret\_Val} \rightarrow \text{Bool})$ 

03	2-5	00	_Analog_Values	6F-3 PN/DP] 🕨 Pro	gram blocks 🔸	Main [OB1]	∎×∎∎				
ю	ы	í I	🖗 👻 🎭 🖿 🚍 🚍 💬 溜 ± 🖀 ± 🚍	😥 🥙 💊 🖑 🧐	l 🚯 l = ½ = 0	× 😤 🔢					
	Main										
		Na	me	Data type	Default value	Comment					
1	-	•	Input								
2	-00	•	Initial_Call	Bool	Initial call of this OB						
З	-	•	Remanence	Bool		=True, if remanent data are available					
4	-	•	Temp								
5	-	•	Motor_speed_monitoring_Ret_Val	Bool 🔳	]						
6		•	<add new=""></add>								
7	-	•	Constant								
8		•	<add new=""></add>								

→ Sélectionnez le titre de bloc de l'OB1 puis cliquez sur <sup>1</sup>/<sub>100</sub>, pour insérer un nouveau réseau 1 avant les autres réseaux.

 $(\rightarrow \overrightarrow{\mathbf{M}})$ 

03	2-5	00	_Analog_Values → CPU1516F [CPU 151	6F-3 PN/DP] > Pro	gram blocks 🔸	Main [OB1] 🛛 🗕 🖬	∎×				
H	ы	ζ.	🖗 👻 🎭 🖿 🚍 🚍 💬 📲 ± 🖴 🖻	😥 🥙 💊 🖑 🖗	1≣ x≡ 6	r 🐨 🔢					
Ч	Main										
Ir	isert	t ne	twork	Data type	Default value	Comment					
1	-00	•	Input								
2		•	Initial_Call	Bool		Initial call of this OB					
З	-00	•	Remanence	Bool		=True, if remanent data are available	ł I				
4	-	•	Temp								
5	-	•	Motor_speed_monitoring_Ret_Val	Bool	)						
6		•	<add new=""></add>								
7	-	•	Constant								
8		•	<add new=""></add>								
	<						>				
-	Blo	ck	title: "Main Program Sweep (Cycle)"				^				
(	Com	me	nt								
•	1	Vet	work 1: Control conveyor motor forwards in a	utomatic mode			=				
	C	om	iment								

→ Faites glisser la fonction "MOTOR\_ SPEEDCONTROL [FC10]" dans le réseau 1 sur la ligne verte.

Vi	å Siemens - G:\Automation\032-500_Analog_Values\032-500_Analog_Values _ 디 🗙											
P	oject Edit View Insert Online Options F 🎦 📮 Save project 📑 💥 🗐 🗎 🗙 🕨	יסד לי <b>ב כ</b> י	ols (~ ±	Wii : T	idow I	Help 🚹 🖳 🎇 💋 Go online 🖉 Go offlir	ne 👍 🖪 🖪 🗡			Totally Integrated Auto	omation PORTAI	L
-	Project tree	∎ ∢	03	32-5	00_An	alog_Values 🕨 CPU1516F [CPU '	1516F-3 PN/DP] 🔸	Prog	ram blocks 🕨	Main [OB1]	_ <b>=</b> = ×	
	Devices											
	1×00	1 🔿	á	ن ا	≤ ⇒ ≥ ≥	* k = = = • • 2 •	<u>⊟ ‱ ∕∾ ⊾ #</u>	QH :	<b>0</b> ≿ I_ %_ /	8, <b>00</b> , 11,		Sul 1
			1	Ma	in				• = = = •			E
÷.	<ul> <li>D 032-500 Analog Values</li> </ul>	^			Name		Data type	r	Default value	Comment		8
Ē	Add new device		1		▼ Inpu	ut	boto type			connent		2
L Bo	H Devices & networks		2			initial_Call	Bool			Initial call of this OB		
l ä	CPU1516F [CPU 1516F-3 PN/DP]		3			 Remanence	Bool			=True, if remanent data a	re available	<b>V.</b>
۲ ۲	Device configuration		4		▼ Tem	1p						1 e
-	🗓 Online & diagnostics		5		•	Motor_speed_monitoring_Ret_Val	Bool					stir
	🔻 🔙 Program blocks		6			<add new=""></add>		_				ē
	💣 Add new block	=	7		▼ Con	stant						
	🌇 Main [OB1]		8		•	<add new=""></add>						
	MOTOR_SPEEDCONTROL [FC10]			<							>	Tas
	MOTOR_AUTO [FB1]		Ŀ	Pla	ok titlo	* "Main Brogram Sweep (Ovcla)"	A				-	ŝ
	MOTOR_AUTO_DB [DB1]		E,	Com	ck uue	. Main Hogram Sweep (Cycle)						
	Technology objects			com	incine							
	External source files		-		Networl	k 1:						F
	PLC tags			C	ommen	t						rar
	Le PLC data types										_	ies
	Watch and force tables			-		MOTOK_SPEEDCONIROL [FCT0]					-	
	Online backups					1045 						
	Traces											

→ Connectez les contacts avec les constantes et les variables globales et locales, comme représenté ci-après.

032-500_Analog_Values → CPU1516F [CPU 1516F-3 PN/DP] → Program blocks → Main [OB1] 🛛 🗖 🖬 🗙										
ю	Ma	in			• = = •					
		Na	me	Data type	Default value	Comment				
1	-00	•	Input	31						
2	-	•	Initial_Call	Bool		Initial call of this OB				
3	-00	•	Remanence	Bool		=True, if remanent data are a	available			
4		•	Temp							
5	-00	•	Motor_speed_monitoring_Ret_Val	Bool 🔳						
6	[	•	<add new=""></add>							
7	-00	•	Constant							
8		•	<add new=""></add>							
	۲						>			
-	Blo	ck	title: "Main Program Sweep (Cycle)"				~			
C	om	me	nt							
•	1	let	twork 1: Speed control analog oputput convey	or motor						
	C	om	iment							
			%FC10							
			"MOTOR_SPEEDCONTROL"							
				#Motor_speed_	-					
				monitoring_Ret	_					
	Ret_Val									
	variable_ %QW64									
			EN speed_	AO — "-U1"						
			15.0 — Setpoint_speed EI	— ov						

→ Modifiez la connexion de la variable de sortie "Conveyor\_motor\_automatic\_mode" dans le réseau 2 sur "-Q3" (Conveyor motor -M1 variable speed (moteur de convoyeur –M1 vitesse variable)), afin que le moteur de convoyeur soit commandé en tenant compte de la vitesse analogique transmise. (→ -Q3)



Utilisation libre pour les instituts publics de formation et de R&D. © Siemens AG 2017. Tous droits réservés. SCE\_FR\_032-500 Analog Values\_S7-1500\_R1703.docx

## 7.6 Enregistrer et compiler le programme

→ Pour enregistrer le projet, sélectionnez le bouton " Save project " dans le menu. Pour compiler tous les blocs, cliquez sur le dossier "Blocs de programme" et dans le menu sur



8	📩 Devices & networks					
리	CPU1516F [CPU 1516F-3 PN/DP]		<b>▼</b>	Network 1: Speed control analog oputput conveyor motor		8
žI	Device configuration	=	C	comment	=	e
	Q Online & diagnostics					stir
	Program blocks			%FC10	1	ē
	📑 Add new block			"MOTOR_SPEEDCONTROL"		
	Hain [OB1]					
	MOTOR_SPEEDCONTROL [FC10]			#Motor_speed_ monitoring_Ret		Tag
	MOTOR_AUTO [FB1]			Ret Val — Val		sks
	MOTOR_AUTO_DB [DB1]			Manipulated		
	Technology objects			variable%QW64		
	External source files			EN speed_AO		5
	PLC tags			15.0 Setpoint_speed ENO		ora
	PLC data types					ries
	Watch and force tables		<b>•</b>	Network 2: Control conveyor motor forwards in automatic mode		Ĩ
	🕨 📴 Online backups				~	
- h	< <u>~</u> m	>	<			
	> Details view			📴 Properties 🚺 Info 🔒 💆 Diagnostics 📑 🖃		
		-	Anim			

→ Les blocs compilés avec succès sont affichés dans la zone "Info" "Compiler".

	Roperties	🗓 Info	i 🖁 🕻	)iagnosti	ics	∎ ■ ▼
General () Cross-references	Compile Syntax					
Show all messages						
Compiling completed (errors: 0; warnin	gs: 0)					
! Path	Description	Go to	?	Errors	Warnings	Time
✓ ▼ CPU1516F		- <b>X</b>		0	0	12:2
< 🔻 Program blocks		<b>N</b>		0	0	12:2
MOTOR_SPEEDCONTRO	Block was successfully compiled.	× .				12:2
Solution (Main (OB1)	Block was successfully compiled.	× .				12:2
<b>v</b>	Compiling completed (errors: 0; warnings: 0)					12:2
<						>

#### 7.7 Charger le programme

→ Une fois la compilation terminée avec succès, le programme créé, y compris la configuration matérielle, peut être chargé dans l'automate comme décrit dans les modules précédents.





#### 7.8 Visualiser les blocs de programme

→ Pour visualiser le programme chargé, le bloc voulu doit être ouvert. Ensuite, vous pouvez

afficher ou masquer la visualisation en cliquant sur l'icône



→ La fonction "MOTOR\_ SPEEDCONTROL" [FC10] appelée dans le bloc d'organisation "Main [OB1]" peut être ouverte et visualisée avec un clic droit (sélectionner "Ouvrir et visualiser") et le code programme de la fonction visualisé.



 $(\rightarrow MOTOR\_SPEEDCONTROL [FC10] \rightarrow Ouvrir et visualiser)$ 

## 7.9 Archivage du projet

- → Pour finir, nous voulons archiver le projet complet. Dans le menu → "Projet", sélectionnez la commande → "Archiver...". Choisissez le dossier d'archivage du projet et enregistrez le projet au format "Archive de projet TIA Portal".
  - $(\rightarrow \text{Projet} \rightarrow \text{Archiver} \rightarrow \text{Archive de projet TIA Portal} \rightarrow 032-500_\text{Analog_Values}...$
  - $\rightarrow$  Enregistrer)



# 8 Liste de contrôle

N٥	Description	Vérifié
1	Compilation réussie et sans message d'erreur	
2	Chargement réussi et sans message d'erreur	
3	Mettre en marche l'installation (-K0 = 1) Vérin rentré / Réponse activée (-B1 = 1) Arrêt d'urgence (-A1 = 1) non activé Mode AUTOMATIQUE (-S0 = 1) Bouton Arrêt Automatique non actionné (-S2 = 1) Actionner brièvement le bouton Démarrage automatique (-S1 = 1) Capteur toboggan affecté activé (-B4 = 1) puis le moteur du convoyeur -M1 vitesse variable se met en marche (-Q3 = 1) et reste actif. La vitesse correspond à la consigne de vitesse dans la plage +/- 50 tr/min	
4	Capteur fin du convoyeur (-B7 = 1) $\rightarrow$ -Q3 = 0 (après 2 secondes	
5	Actionner brièvement le bouton arrêt automatique (-S2 = 0) $\rightarrow$ -Q3 = 0	
6	Activer l'arrêt d'urgence (-A1 = 0) $\rightarrow$ -Q3 = 0	
7	Mode manuel (-S0 = 0) $\rightarrow$ -Q3 = 0	
8	Mettre l'installation à l'arrêt (-K0 = 0) $\rightarrow$ -Q3 = 0	
9	Vérin non rentré (-B1 = 0) $\rightarrow$ -Q3 = 0	
10	Le projet a été archivé avec succès	

# 9 Exercice

## 9.1 Énoncé du problème - exercice

Dans cet exercice, la fonction "MOTOR\_SPEEDMONITORING" [FC11] (surveillance de la vitesse du moteur) doit être créée en supplément.

La valeur réelle est mise à disposition sous forme de valeur analogique -B8 (Capteur Valeur réelle de vitesse du moteur +/-10V correspondent à +/- 50 tr/min) et interrogée à une entrée de la fonction "MOTOR\_SPEEDMONITORING" [FC11]. Le type de données est ici un nombre entier 16 bits (Int).

Dans la fonction, cette valeur de mesure de vitesse est d'abord normalisée à la plage +/-1 en tant que nombre à virgule flottante 32 bits (Real).

Ensuite, la valeur de mesure de vitesse normalisée sera dénormalisée en tours par minute (plage : +/- 50 tr/min) sous forme de nombre à virgule flottante 32 bits (Real) et mise à disposition sur une sortie.

Les quatre valeurs limites suivantes peuvent être transmises sous forme de nombre à virgule flottante 32 bits (Real) sur les entrées de bloc, aux fins de la surveillance dans la fonction :

Vitesse > seuil de vitesse erreur max

Vitesse > seuil de vitesse avertissement max

Vitesse < seuil de vitesse avertissement min

Vitesse < seuil de vitesse erreur min

Si un seuil est dépassé par le haut ou par le bas, la valeur TRUE (1) est affectée au bit de sortie correspondant.

Si un défaut se produit, la coupure de sécurité du bloc fonctionnel "MOTOR\_AUTO" [FB1] doit être activée.

## 9.2 Schéma technologique

Vous voyez ici le schéma technologique de l'énoncé du problème.



Figure 3 : Schéma technologique

Schalter der Sortieranlage Switches of sorting station	Automatikbetrieb Automatic mode	Handbetrieb / Manual mode -S3 Tippbetrieb -M1 vorwärts/ Manual -M1 forwards
-Q0 Hauptschalter/Main switch -P4 aktiviert/active -A1 NOTHALT/Emergency stop	-S1 Start/start	-S4 Tippbetrieb -M1 rückwärts/ Manual -M1 backwards -P7 ausgefahren/extended
-P2 Handimanual -P3 Auto/auto		Cylinder -M4 extend -P6 eingefahren/retracted -S5 Zylinder -M4 einfahren/ cylinder -M4 retract

Figure 4 : Pupitre de commande

## 9.3 Tableau d'affectation

DI	Туре	Désignation	Fonction	NF/NO
E 0.0	BOOL	-A1	Alarme Arrêt d'urgence ok	NF
E 0.1	BOOL	-K0	Installation "Marche"	NO
E 0.2	BOOL	-S0	Commutateur mode manuel (0) / automatique (1)	Manuel = 0 Auto=1
E 0.3	BOOL	-S1	Bouton poussoir démarrage automatique	NO
E 0.4	BOOL	-S2	Bouton poussoir arrêt automatique	NF
E 0.5	BOOL	-B1	Capteur vérin -M4 rentré	NO
E 1.0	BOOL	-B4	Capteur toboggan affecté	NO
E 1.3	BOOL	-B7	Capteur de pièce en fin de convoyeur	NO
EW64	BOOL	-B8	Capteur Mesure vitesse du moteur +/-10V correspondent à +/- 50 tr/min	

Les signaux suivants seront nécessaires pour cette tâche, au titre d'opérandes globaux.

DO	Туре	Désignation	Fonction	
A 0.2	BOOL	-Q3	Moteur du convoyeur -M1 vitesse variable	
AW 64	BOOL	-U1	Valeur de réglage de la vitesse du moteur dans les 2 directions +/-10V correspondent à +/- 50 tr/min	

#### Légende de la liste d'affectation

- DI Entrée TOR DQ Sortie TOR
- Al Entrée analogique AQ Sortie analogique

Sortie

- Entrée Q
- NF Normalement fermé (contact à ouverture)
- NO Normalement ouvert (contact à fermeture)

#### 9.4 Planification

L

Planifiez seul la réalisation de l'énoncé.

## 9.5 Liste de contrôle – Exercice

Nº	Description	Vérifié	
1	Compilation réussie et sans message d'erreur		
2	Chargement réussi et sans message d'erreur		
3	Mettre en marche l'installation (-K0 = 1) Vérin rentré / Réponse activée (-B1 = 1) Arrêt d'urgence (-A1 = 1) non activé Mode AUTOMATIQUE (-S0 = 1) Bouton Arrêt Automatique non actionné (-S2 = 1) Actionner brièvement le bouton Démarrage automatique (-S1 = 1) Capteur toboggan affecté activé (-B4 = 1) puis le moteur du convoyeur -M1 vitesse variable se met en marche (-Q3 = 1) et reste actif. La vitesse correspond à la consigne de vitesse dans la plage +/- 50 tr/min		
4	Capteur fin du convoyeur (-B7 = 1) $\rightarrow$ -Q3 = 0 (après 2 secondes		
5	Actionner brièvement le bouton arrêt automatique $(-S2 = 0) \rightarrow -Q3 = 0$		
6	Activer l'arrêt d'urgence (-A1 = 0) $\rightarrow$ -Q3 = 0		
7	Mode manuel (-S0 = 0) $\rightarrow$ -Q3 = 0		
8	Mettre l'installation à l'arrêt (-K0 = 0) $\rightarrow$ -Q3 = 0		
9	Vérin non rentré (-B1 = 0) $\rightarrow$ -Q3 = 0		
10	Vitesse > seuil de vitesse erreur max $\rightarrow$ -Q3 = 0		
11	Vitesse < seuil de vitesse erreur min $\rightarrow$ -Q3 = 0		
12	Le projet a été archivé avec succès		

# **10Informations complémentaires**

Des informations complémentaires vous sont proposées afin de vous aider à vous exercer ou à titre d'approfondissement, par ex. : Getting Started, vidéos, didacticiels, applis, manuels, guides de programmation et logiciel/firmware d'évaluation sous le lien suivant :

www.siemens.com/sce/s7-1500