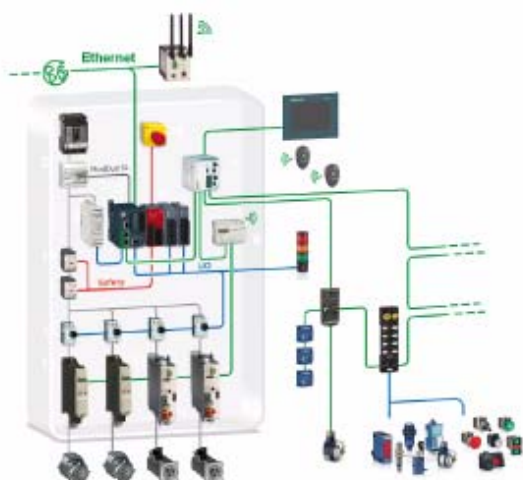


# SoMachine Scrutateur d'E/S Modbus TCP Guide de l'utilisateur

09/2014



---

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.

Aucune partie de ce document ne peut être reproduite sous quelque forme ou par quelque moyen que ce soit, électronique, mécanique ou photocopie, sans autorisation préalable de Schneider Electric.

Toutes les réglementations de sécurité pertinentes locales doivent être observées 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.

© 2014 Schneider Electric. Tous droits réservés.

---

# Table des matières

---



	<b>Consignes de sécurité</b> .....	<b>5</b>
	<b>A propos de ce manuel</b> .....	<b>7</b>
<b>Chapitre 1</b>	<b>Scrutateur d'E/S Modbus TCP - Présentation</b> .....	<b>9</b>
	Scrutateur d'E/S Modbus TCP - Présentation .....	<b>10</b>
	Architecture .....	<b>12</b>
	Principes .....	<b>14</b>
<b>Chapitre 2</b>	<b>Installation du réseau</b> .....	<b>17</b>
	Planification du réseau .....	<b>18</b>
	Configuration d'adresse IP .....	<b>20</b>
	Tests du réseau .....	<b>23</b>
<b>Chapitre 3</b>	<b>Scrutateur d'E/S Modbus TCP - Configuration</b> .....	<b>25</b>
	Ajout d'un esclave au scrutateur d'E/S Modbus TCP .....	<b>26</b>
	Configuration d'un Scrutateur d'E/S Modbus TCP .....	<b>27</b>
	Configuration d'un module d'E/S distribuées Advantys OTB sur le Scrutateur d'E/S Modbus TCP .....	<b>29</b>
	Configuration d'un esclave prédéfini sur le Scrutateur d'E/S Modbus TCP .....	<b>32</b>
	Configuration d'un équipement générique sur le Scrutateur d'E/S Modbus TCP .....	<b>34</b>
<b>Chapitre 4</b>	<b>Scrutateur d'E/S Modbus TCP - Fonctionnement</b> . . . .	<b>37</b>
	Scrutateur d'E/S Modbus TCP - Vérification des ressources .....	<b>38</b>
	Scrutateur d'E/S Modbus TCP - Modes de fonctionnement .....	<b>40</b>
	Interface d'application .....	<b>44</b>
<b>Chapitre 5</b>	<b>Scrutateur d'E/S Modbus TCP - Maintenance</b> .....	<b>47</b>
	Diagnostics : Mode en ligne de SoMachine .....	<b>48</b>
	Diagnostics : Serveur Web .....	<b>52</b>
	Dépannage .....	<b>54</b>
<b>Annexes</b>	.....	<b>55</b>
<b>Annexe A</b>	<b>Scrutateur d'E/S Modbus TCP - Fonctions</b> .....	<b>57</b>
	IOS_GETSTATE : Lit l'état du Scrutateur d'E/S Modbus TCP .....	<b>58</b>
	IOS_START : Lance le Scrutateur d'E/S Modbus TCP .....	<b>59</b>
	IOS_GETHEALTH : Lit la valeur du bit de validité .....	<b>60</b>
	IOS_STOP : Arrête le Scrutateur d'E/S Modbus TCP .....	<b>61</b>
	CONFIGURE_OTB : Envoie la configuration logicielle d'un équipement Advantys OTB .....	<b>62</b>

---

<b>Annexe B</b>	<b>Scrutateur d'E/S Modbus TCP - Types de données . .</b>	<b>65</b>
	losStateCodes : Valeurs d'état Scrutateur d'E/S Modbus TCP . . . . .	<b>66</b>
	CommunicationErrorCodes : Codes d'erreur détectée . . . . .	<b>67</b>
	configurationOTBErrorCodes : Codes d'erreur détectée dans la configuration OTB . . . . .	<b>68</b>
<b>Annexe C</b>	<b>Représentation des fonctions et blocs fonction . . . .</b>	<b>69</b>
	Différences entre une fonction et un bloc fonction . . . . .	<b>70</b>
	Utilisation d'une fonction ou d'un bloc fonction en langage IL . . . . .	<b>71</b>
	Utilisation d'une fonction ou d'un bloc fonction en langage ST . . . . .	<b>75</b>
<b>Glossaire</b>	. . . . .	<b>79</b>
<b>Index</b>	. . . . .	<b>83</b>

# 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 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

Utilisez ce document pour :

- Planifier votre réseau Scrutateur d'E/S Modbus TCP
- Installer et configurer votre réseau Scrutateur d'E/S Modbus TCP
- Utiliser et gérer votre réseau Scrutateur d'E/S Modbus TCP

**NOTE** : Lisez attentivement ce document et tous les documents associés avant de procéder à l'installation, l'utilisation ou la maintenance de votre contrôleur.

### Champ d'application

Le présent document a été actualisé suite au lancement du module complémentaire Scrutateur d'E/S Modbus TCP de SoMachine V4.1.

Les caractéristiques techniques des équipements décrits dans ce manuel sont également disponibles en ligne.

Les caractéristiques présentées dans ce manuel devraient être identiques à celles fournies en ligne. Toutefois, en application de notre politique d'amélioration continue, nous pouvons être amenés à réviser le contenu du document afin de le rendre plus clair et plus précis. Si vous constatez une différence entre le manuel et les informations fournies en ligne, utilisez ces dernières en priorité.

### Document(s) à consulter

Titre de documentation	Référence
Modicon M251 Logic Controller - Guide de programmation	EIO0000001462 (ENG), EIO0000001463 (FRE), EIO0000001464 (GER), EIO0000001465 (SPA), EIO0000001466 (ITA), EIO0000001467 (CHS)
SoMachine - Guide de programmation	EIO0000000067 (ENG); EIO0000000069 (FRE); EIO0000000068 (GER); EIO0000000071 (SPA); EIO0000000070 (ITA); EIO0000000072 (CHS)
Essential Guide: Networks, connectivity and Web servers	DIA6ED2130205EN (ENG)

Vous pouvez télécharger ces publications et autres informations techniques depuis notre site web à l'adresse : [www.schneider-electric.com](http://www.schneider-electric.com).

## Information spécifique au produit

### AVERTISSEMENT

#### PERTE DE CONTROLE

- Le concepteur d'un circuit de commande doit tenir compte des modes de défaillance potentiels des canaux de commande et, pour certaines fonctions de commande critiques, prévoir un moyen d'assurer la sécurité en maintenant un état sûr pendant et après la défaillance. Par exemple, l'arrêt d'urgence, l'arrêt en cas de surcourse, la coupure de courant et le redémarrage sont des fonctions de commande cruciales.
- Des canaux de commande séparés ou redondants doivent être prévus pour les fonctions de commande critiques.
- Les liaisons de communication peuvent faire partie des canaux de commande du système. Une attention particulière doit être prêtée aux implications des délais de transmission non prévus ou des pannes de la liaison.
- Respectez toutes les réglementations de prévention des accidents ainsi que les consignes de sécurité locales.<sup>1</sup>
- 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.**

<sup>1</sup> 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 INATTENDU DE L'ÉQUIPEMENT

- 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

## Scrutateur d'E/S Modbus TCP - Présentation

---

### Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
Scrutateur d'E/S Modbus TCP - Présentation	10
Architecture	12
Principes	14

## Scrutateur d'E/S Modbus TCP - Présentation

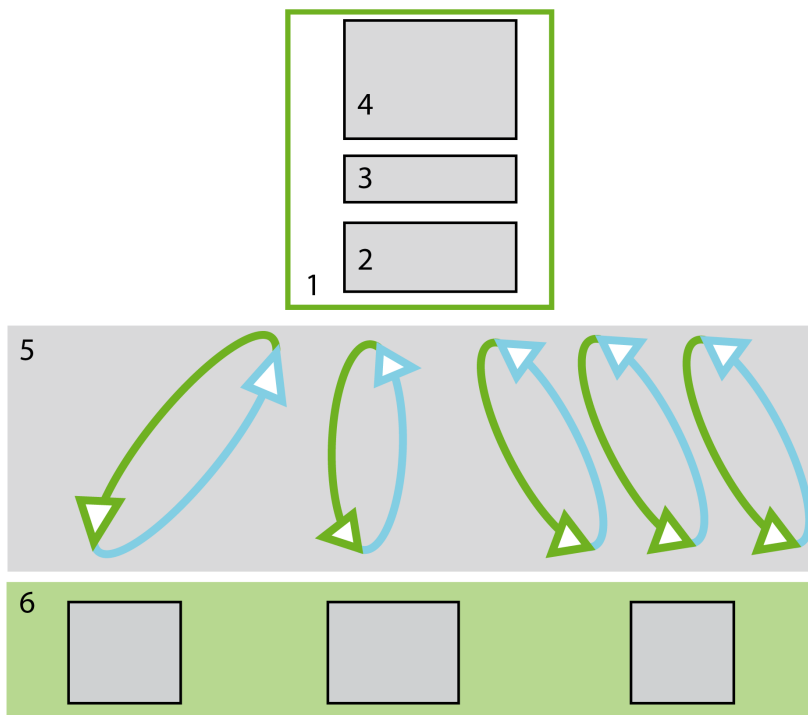
### Présentation

Le Scrutateur d'E/S Modbus TCP est un service basé sur Ethernet qui interroge en permanence des équipements esclaves pour échanger des données et des informations d'état et de diagnostic. Ce processus surveille les entrées et contrôle les sorties des équipements esclaves.

Le Scrutateur d'E/S Modbus TCP s'appuie sur la norme Modbus TCP qui s'articule autour d'un modèle réseau maître/esclave. Le contrôleur est l'unique maître.

La communication avec les esclaves est assurée par des voies Modbus TCP (*voir page 14*).

### Principe



- 1 Contrôleur
- 2 Images d'E/S
- 3 Interface d'application (*voir page 44*)
- 4 Application
- 5 Voies Modbus (*voir page 14*)
- 6 Equipements esclaves (*voir page 14*)

### Architecture du système

Le Scrutateur d'E/S Modbus TCP repose sur les éléments suivants :

- Réseau Ethernet comprenant le contrôleur, les esclaves et les équipements d'infrastructure (*voir page 12*)
- Configuration logicielle (*voir page 13*)

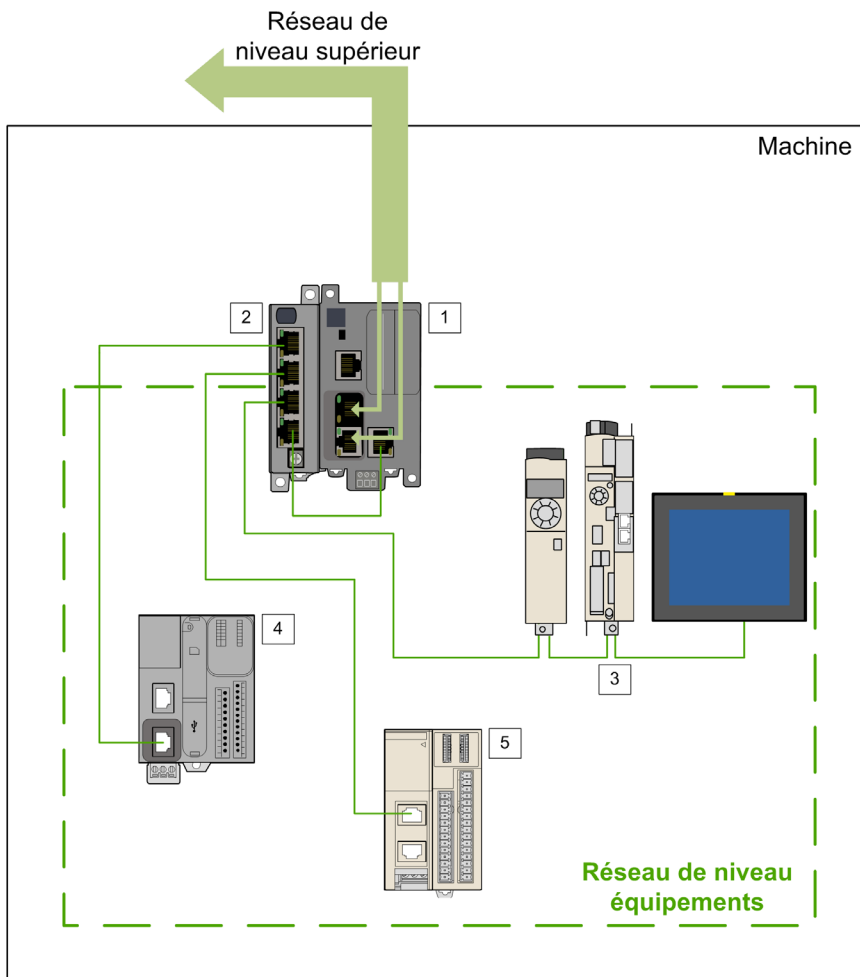
### Compatibilité des contrôleurs

Le service Scrutateur d'E/S Modbus TCP est disponible sur le contrôleur TM251MESE.

## Architecture

### Réseau Ethernet

La figure suivante présente une architecture Scrutateur d'E/S Modbus TCP typique.



- 1 Contrôleur / maître Modbus
- 2 TM4ES4 utilisé comme commutateur Ethernet autonome
- 3 Esclaves enchaînés
- 4 Esclave Modbus
- 5 Ilot d'E/S

Le contrôleur est connecté au réseau de niveau supérieur ainsi qu'au réseau de niveau équipements.

Le réseau de niveau équipements est contrôlé par le Scrutateur d'E/S Modbus TCP.

Le contrôleur peut être utilisé comme passerelle (*voir page 20*) entre les deux réseaux.

### Configuration logicielle

La zone réseau des esclaves et de chaque I/O est configurée par logiciel :

- La configuration des communications définit l'adressage et les périodes de communication.
- La configuration des équipements définit le comportement de ces derniers.

La configuration des esclaves permet l'attribution de variables pour optimiser la surveillance.

La configuration des E/S ajuste la qualité de la surveillance en fonction de la bande passante du réseau.

Vous pouvez définir plusieurs paramètres (*voir page 15*) pour optimiser les performances.

### Services facultatifs

Le Scrutateur d'E/S Modbus TCP peut être associé à plusieurs services facultatifs :

- DHCP : Le serveur DHCP (Dynamic Host Configuration Protocol) affecte une adresse IP à un équipement esclave qui en fait la demande.
- FDR : Le serveur FDR (Fast Device Replacement) configure un équipement distant qui a été remplacé et qui est contrôlé par le Scrutateur d'E/S Modbus TCP sans arrêter l'application.
- Serveur Web (*voir page 52*).

## Principes

### Présentation

Le Scrutateur d'E/S Modbus TCP lit les entrées et écrit les sorties des équipements esclaves.

La communication entre le Scrutateur d'E/S Modbus TCP et les équipements esclaves est assurée à l'aide de voies Modbus.

Dans le Scrutateur d'E/S Modbus TCP, la communication est configurée avec le logiciel SoMachine.

### Types d'esclave

Il existe trois types d'esclave Scrutateur d'E/S Modbus TCP dans le logiciel SoMachine :

- Les équipements esclaves **Advantys OTB** sont utilisés pour les E/S numériques et analogiques distantes.

Utilisez le logiciel SoMachine pour la configuration spécifique de l'équipement et des modules d'E/S associés.

- Les équipements esclaves **prédéfinis** sont des appareils Modbus courants qui sont fournis avec un ensemble prédéfini de paramètres de communication.

Utilisez un logiciel dédié et/ou une IHM locale pour configurer ces équipements. La technologie FDT/DTM permet de configurer dans SoMachine des équipements esclaves prédéfinis avec des paramètres avancés (voir Device Type Manager (DTM) - Guide de l'utilisateur).

- Les équipements esclaves **génériques** sont utilisés pour tous les autres appareils esclaves Modbus.

La totalité de la configuration de ces équipements est effectuée à l'aide d'un logiciel tiers et/ou d'une IHM locale. La technologie FDT/DTM permet d'en configurer certains dans SoMachine (voir Device Type Manager (DTM) - Guide de l'utilisateur).

### Voie Modbus

Une voie Modbus véhicule une requête Modbus entre le maître et un esclave.

Les esclaves Advantys OTB et prédéfinis utilisent une voie par équipement. Cette voie est configurée à l'aide du logiciel SoMachine.

Dans le cas d'un équipement esclave générique, il est possible d'utiliser plusieurs voies pour envoyer diverses requêtes au même équipement.

## Paramètres de configuration des communications

Configurez chaque esclave du réseau Scrutateur d'E/S Modbus TCP à l'aide du logiciel SoMachine.

Le tableau suivant présente les paramètres de configuration des communications :

Paramètre	Description
Adresse IP	Adresse IP de l'esclave dans le réseau Ethernet.
Timeout de validité	Durée exprimée en millisecondes (ms). Si le Scrutateur d'E/S Modbus TCP ne détecte pas de réponse de l'esclave à l'expiration de ce délai, une situation d'erreur se produit.
Vitesse de répétition	Durée exprimée en millisecondes (ms). Il s'agit du délai entre deux envois successifs d'une requête. Sa valeur doit être inférieure au timeout de validité.
ID de voie	Identificateur unique d'une voie. Cette valeur est créée automatiquement par le logiciel SoMachine lors de l'ajout d'une nouvelle voie. Vous pouvez la lire dans l'onglet <b>Configuration de voie Modbus TCP</b> ou, pour l'esclave Advantys OTB, dans l'onglet <b>Configuration esclave Modbus TCP</b> .



---

# Chapitre 2

## Installation du réseau

---

### Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
Planification du réseau	18
Configuration d'adresse IP	20
Tests du réseau	23

## Planification du réseau

### Objectif

Un réseau planifié permet d'augmenter l'efficacité et de réduire les temps et les coûts d'installation. Les interfaces matérielles (commutateurs, câbles, ports) doivent être conçues à l'avance pour planifier le réseau.

### Conception du réseau

Pour concevoir et planifier le réseau Scrutateur d'E/S Modbus TCP, consultez la documentation correspondante, notamment Media Planning and Installation Manual de l'ODVA. Vous pouvez télécharger ce manuel à partir du [site Web de l'ODVA](#).

### Types de commutateur

En fonction des besoins propres à votre réseau, utilisez le type de commutateur approprié :

Pour les besoins suivants	Prévoyez d'utiliser
Diagnostics réseau et informations de fonctionnement	Commutateurs gérables
Disponibilité des communications en cas de perte de connexion physique	Commutateurs redondants
Réseau longue portée (fibre optique)	Commutateur avec connecteur SC duplex

**NOTE** : N'utilisez pas de concentrateur pour configurer un réseau Scrutateur d'E/S Modbus TCP. Pour plus d'informations sur les commutateurs, reportez-vous au document **Essential Guide: Networks, connectivity and Web servers**.

## Types de câble

Les tableaux suivants présentent les références de câble pouvant être utilisées dans le réseau.

Pour plus d'informations sur les câbles, reportez-vous au document  
**Essential Guide: Networks, connectivity and Web servers.**

Dans une installation standard, vous pouvez utiliser les câbles suivants :

Référence	Description	Détails	Longueur
490NTW000**	Câble Ethernet blindé pour connexions ETDD	Câble standard équipé de connecteurs RJ45 à chaque extrémité pour ETDD. Conformité CE	2, 5, 12, 40 ou 80 m (6.56, 16.4, 39.37, 131.23 ou 262.47 ft)
490NTW000**U		Câble standard équipé de connecteurs RJ45 à chaque extrémité pour ETDD. Conformité UL	2, 5, 12, 40 ou 80 m (6.56, 16.4, 39.37, 131.23 ou 262.47 ft)
TCSECE3M3M**S4		Câble pour environnement exigeant, équipé de connecteurs RJ45 à chaque extrémité. Conformité CE	1, 2, 3, 5 ou 10 m (3.28, 6.56, 9.84, 16.4, 32.81 ft)
TCSECU3M3M**S4		Câble pour environnement exigeant, équipé de connecteurs RJ45 à chaque extrémité. Conformité UL	1, 2, 3, 5 ou 10 m (3.28, 6.56, 9.84, 16.4, 32.81 ft)

Dans le cas d'un réseau à fibre optique, vous pouvez utiliser les câbles suivants :

Référence	Description	Détails	Longueur
490NOC00005	Câble optique en fibre de verre pour connexions ETDD	1 connecteur SC 1 connecteur MT-RJ	5 m (16.4 ft)
490NOT00005		1 connecteur ST (BFOC) 1 connecteur MT-RJ	5 m (16.4 ft)
490NOR00003		2 connecteurs MT-RJ	3 m (9.8 ft)
490NOR00005		2 connecteurs MT-RJ	5 m (16.4 ft)

## Configuration d'adresse IP

### Conditions requises

Chaque équipement du réseau Scrutateur d'E/S Modbus TCP se procure sa propre adresse IP. Toutes les adresses IP doivent être uniques.

**NOTE :** Affectez des adresses IP de classe C pour le réseau Scrutateur d'E/S Modbus TCP.

Configurez les adresses IP en deux étapes :

Etape	Description
1	<p>A l'aide du logiciel SoMachine, configurez le port du contrôleur qui prend en charge le Scrutateur d'E/S Modbus TCP :</p> <ul style="list-style-type: none"> <li>● Adresse IP,</li> <li>● Masque de sous-réseau,</li> <li>● Passerelle par défaut.</li> </ul> <p><b>NOTE :</b> Utilisez uniquement des adresses fixes.</p> <p><b>NOTE :</b> N'utilisez pas le fichier de post-configuration pour l'adressage des ports du contrôleur.</p>
2	<p>Configurez chaque équipement esclave du réseau Scrutateur d'E/S Modbus TCP :</p> <ul style="list-style-type: none"> <li>● Adresse IP,</li> <li>● Masque de sous-réseau,</li> <li>● Passerelle par défaut.</li> </ul>

### Affectation d'adresse aux équipements esclaves

Dans le réseau Scrutateur d'E/S Modbus TCP, affectez l'adresse IP des équipements esclaves à l'aide d'une des méthodes suivantes (en fonction du type d'équipement) :

- Via un serveur DHCP, pour gérer toutes les adresses IP du réseau Scrutateur d'E/S Modbus TCP à partir du logiciel SoMachine ou si vous avez besoin du service FDR
- Via un logiciel tiers ou une IHM locale
- Via une configuration de paramètres avancés sur ligne série Modbus (Modbus SL) via la technologie FDT/DTM (voir Device Type Manager (DTM) - Guide de l'utilisateur).

### Service FDR

Certains équipements esclaves prennent en charge le service de remplacement rapide (FDR - Fast Device Replacement).

Le service FDR stocke des paramètres réseau et d'exploitation concernant les équipements du réseau. Si un appareil est remplacé, ce service configure automatiquement l'appareil de remplacement avec les paramètres de l'appareil déposé.

Pour configurer ce service dans l'esclave, reportez-vous à la documentation relative à l'équipement esclave.

## Adresse IP maître d'esclave

Certains appareils esclaves sont configurés avec un paramètre **Adresse IP maître** qui fait qu'un seul contrôleur maître y a accès.

Si l'appareil...	Alors...
Est configuré pour utiliser le Scrutateur d'E/S Modbus TCP	Configurez le paramètre <b>Adresse IP maître</b> à l'intérieur de l'appareil. Voir ci-après.
N'est pas configuré pour utiliser le Scrutateur d'E/S Modbus TCP	Utilisez 0 . 0 . 0 . 0 comme valeur du paramètre <b>Adresse IP maître</b> dans l'appareil.

Le paramètre **Adresse IP maître** de l'esclave doit désigner l'adresse IP du contrôleur qui prend en charge le Scrutateur d'E/S Modbus TCP (port **Ethernet 2**).

Pour configurer ce paramètre dans l'équipement esclave, reportez-vous à la documentation de cet équipement.

## Paramètre de passerelle des esclaves

Le paramètre de passerelle des équipements esclaves doit être l'adresse IP du port Ethernet du contrôleur qui prend en charge le Scrutateur d'E/S Modbus TCP (**Ethernet 2**).

Un outil de configuration doit accéder aux esclaves pour définir leurs paramètres.

Si l'outil de configuration...	Alors...
Est connecté sur le réseau de niveau supérieur	Mettez à jour le paramètre de passerelle de l'esclave (voir plus bas).
Est connecté sur le réseau de niveau équipements	Le paramètre de passerelle n'est pas utilisé.
Utilisez un protocole autre que TCP/IP.	Le paramètre de passerelle n'est pas utilisé.

Pour configurer ce paramètre dans l'équipement esclave, reportez-vous à la documentation de cet équipement.

**NOTE** : Si le service DHCP est utilisé pour l'adressage des équipements esclaves, le paramètre de passerelle est défini dans la table DHCP du contrôleur.

## Routage par PC

Le PC qui prend en charge l'outil de configuration doit être configuré pour communiquer avec les équipements esclaves.

Si l'esclave est configuré...	Alors...
En tant qu'esclave prédéfini via des paramètres avancés (FDT/DTM)	Aucun paramétrage informatique n'est nécessaire. <b>NOTE</b> : La configuration de l'ordinateur n'est pas modifiée.
Via un autre outil	Mettez à jour la table de routage du PC (voir plus bas).

Pour mettre à jour la table de routage du PC, arrêtez toute communication du PC vers le contrôleur et/ou d'autres équipements. Ensuite, accédez à la ligne de commande Windows et exécutez la commande `route ADD Destination MASK Subnet_Mask Gateway`.

Le tableau suivant présente les paramètres à mettre à jour dans cette syntaxe de commande :

Paramètre	Valeur
Destination	Adresse IP du réseau Scrutateur d'E/S Modbus TCP
Subnet_Mask	Masque de sous-réseau du réseau Scrutateur d'E/S Modbus TCP.
Gateway	Adresse du port du contrôleur qui est connecté au réseau de niveau supérieur.

Pour vérifier ces paramètres à partir d'une invite de commande Windows, exécutez la commande `route PRINT`.

Pour supprimer ce routage à partir du PC, exécutez la commande `route DELETE Destination` sur la ligne de commande Windows, `Destination` étant l'adresse IP du réseau Scrutateur d'E/S Modbus TCP entrée antérieurement.

## Tests du réseau

### Objectif

Avant d'utiliser le Scrutateur d'E/S Modbus TCP, testez le réseau.

Vérifiez que les conditions suivantes sont satisfaites :

- La configuration d'adresse de chaque équipement est conforme à la planification.
- Chaque équipement est correctement câblé.

Quelques méthodes de test courantes sont présentées ci-dessous.

### Voyant d'état

En fonction des équipements, vérifiez que les voyants d'état indiquent un câblage correct.

### Vérification par ordinateur

A l'aide d'un ordinateur, vérifiez que chaque équipement esclave est connecté et adressé :

Étape	Action
1	Connectez l'ordinateur dans le réseau Scrutateur d'E/S Modbus TCP.
2	Accédez à l'invite de commande.
3	Utilisez une commande <code>ping xxx.xxx.xxx.xxx</code> pour joindre chaque esclave. xxx.xxx.xxx.xxx = adresse IP de l'esclave à tester. <b>NOTE</b> : La commande <code>ping -h</code> affiche l'aide relative à la commande <code>ping</code> .

### Vérification à l'aide d'un serveur Web

A l'aide du serveur Web du contrôleur, vérifiez que ce dernier peut communiquer avec chaque équipement esclave :

Étape	Action
1	Accédez au serveur Web du contrôleur.
2	Ouvrez la page <b>Diagnostics - Ethernet</b> .
3	Utilisez la fonction ( <i>voir page 52</i> ) <b>Remote Ping Service</b> sur chaque équipement esclave.



---

# Chapitre 3

## Scrutateur d'E/S Modbus TCP - Configuration

---

### Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
Ajout d'un esclave au scrutateur d'E/S Modbus TCP	26
Configuration d'un Scrutateur d'E/S Modbus TCP	27
Configuration d'un module d'E/S distribuées Advantys OTB sur le Scrutateur d'E/S Modbus TCP	29
Configuration d'un esclave prédéfini sur le Scrutateur d'E/S Modbus TCP	32
Configuration d'un équipement générique sur le Scrutateur d'E/S Modbus TCP	34

## Ajout d'un esclave au scrutateur d'E/S Modbus TCP

### Présentation

Cette section explique comment ajouter un esclave au **Scrutateur d'E/S Modbus TCP**.

Ces esclaves relèvent de 3 catégories :

- équipements prédéfinis pour équipements Modbus Schneider Electric (ATV, LXM et ZBRN),
- Advantys OTB pour module Modbus TCP avec E/S configurables,
- équipements génériques pour tous les autres esclaves Modbus TCP.

Pour chaque équipement générique, vous devez définir les requêtes Modbus à lui envoyer en ajoutant des voies. Une voie correspond à une requête Modbus et peut avoir sa propre vitesse de répétition.

### Ajout d'un esclave au scrutateur d'E/S Modbus TCP

Pour ajouter un esclave au **Scrutateur d'E/S Modbus TCP**, sélectionnez l'équipement choisi dans le **Catalogue de matériels**, faites-le glisser dans l'arborescence **Equipements** et déposez-le sur le nœud **Ethernet\_2** de l'arborescence **Equipements**.

Pour plus d'informations sur l'ajout d'un esclave au projet, reportez-vous aux rubriques suivantes :

- Utilisation de la Méthode du Glisser-Déposer (*voir SoMachine, Guide de programmation*)
- Utilisation du Menu contextuel ou du bouton Plus (*voir SoMachine, Guide de programmation*)

## Configuration d'un Scrutateur d'E/S Modbus TCP

### Conditions requises

Avant de configurer le Scrutateur d'E/S Modbus TCP :

- Définissez l'adresse IP du port Ethernet 2 en **mode fixe**. Elle doit être différente de 0.0.0.0.
- Les équipements connectés doivent appartenir au même sous-réseau que le port Ethernet 2.

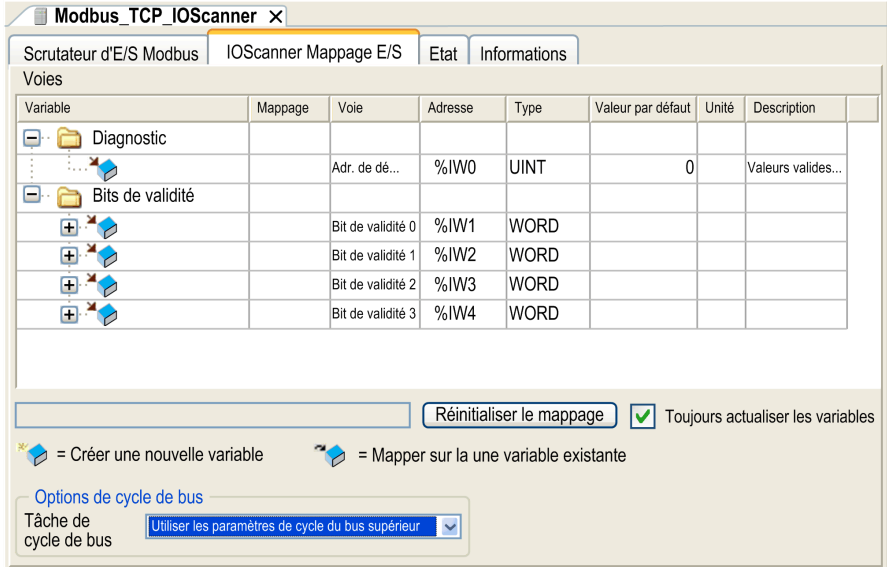
Pour plus d'informations sur l'adresse IP, consultez la section Configuration Ethernet (voir *Modicon M251 Logic Controller, Guide de programmation*).

### Ajout d'un Scrutateur d'E/S Modbus TCP

Le nœud Scrutateur d'E/S Modbus TCP est créé automatiquement lorsqu'un esclave est ajouté sur le nœud **Ethernet 2** (voir page 26).

### Configuration d'un Scrutateur d'E/S Modbus TCP

Pour configurer un Scrutateur d'E/S Modbus TCP, procédez comme suit :

Etape	Action																																																								
1	Dans l'arborescence <b>Equipements</b> , double-cliquez sur <b>Modbus_TCP_IOScanner</b> . <b>Résultat</b> : la fenêtre de configuration s'affiche.																																																								
2	<p>Sélectionnez l'onglet <b>IOScanner Mappage E/S</b>.</p>  <p>Voies</p> <table border="1"> <thead> <tr> <th>Variable</th> <th>Mappage</th> <th>Voie</th> <th>Adresse</th> <th>Type</th> <th>Valeur par défaut</th> <th>Unité</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>Diagnostic</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Bits de validité</td> <td></td> <td>Adr. de dé...</td> <td>%IW0</td> <td>UINT</td> <td>0</td> <td></td> <td>Valeurs valides...</td> </tr> <tr> <td>+</td> <td></td> <td>Bit de validité 0</td> <td>%IW1</td> <td>WORD</td> <td></td> <td></td> <td></td> </tr> <tr> <td>+</td> <td></td> <td>Bit de validité 1</td> <td>%IW2</td> <td>WORD</td> <td></td> <td></td> <td></td> </tr> <tr> <td>+</td> <td></td> <td>Bit de validité 2</td> <td>%IW3</td> <td>WORD</td> <td></td> <td></td> <td></td> </tr> <tr> <td>+</td> <td></td> <td>Bit de validité 3</td> <td>%IW4</td> <td>WORD</td> <td></td> <td></td> <td></td> </tr> </tbody> </table> <p>Réinitialiser le mappage <input checked="" type="checkbox"/> Toujours actualiser les variables</p> <p>* = Créer une nouvelle variable    = Mapper sur la une variable existante</p> <p>Options de cycle de bus Tâche de cycle de bus: Utiliser les paramètres de cycle du bus supérieur</p>	Variable	Mappage	Voie	Adresse	Type	Valeur par défaut	Unité	Description	Diagnostic								Bits de validité		Adr. de dé...	%IW0	UINT	0		Valeurs valides...	+		Bit de validité 0	%IW1	WORD				+		Bit de validité 1	%IW2	WORD				+		Bit de validité 2	%IW3	WORD				+		Bit de validité 3	%IW4	WORD			
Variable	Mappage	Voie	Adresse	Type	Valeur par défaut	Unité	Description																																																		
Diagnostic																																																									
Bits de validité		Adr. de dé...	%IW0	UINT	0		Valeurs valides...																																																		
+		Bit de validité 0	%IW1	WORD																																																					
+		Bit de validité 1	%IW2	WORD																																																					
+		Bit de validité 2	%IW3	WORD																																																					
+		Bit de validité 3	%IW4	WORD																																																					

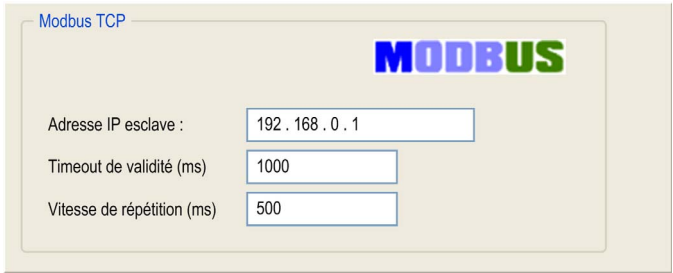
Etape	Action
3	<p>Sélectionnez une option dans la liste <b>Tâche de cycle de bus</b> :</p> <ul style="list-style-type: none"><li>● <b>Utiliser les paramètres de cycle du bus supérieur</b> (option par défaut),</li><li>● <b>MAST</b> ou</li><li>● Une tâche existante du projet.</li></ul> <p><b>NOTE</b> : Le paramètre <b>Tâche de cycle de bus</b> indiqué dans l'éditeur de mappage d'E/S de l'équipement qui contient le Scrutateur d'E/S Modbus TCP définit la tâche responsable de l'actualisation des images d'E/S (%QW, %IW). Ces images d'E/S correspondent à la requête Modbus envoyée aux esclaves Modbus et aux bits de validité.</p>

**NOTE** : Lorsque le Scrutateur d'E/S Modbus TCP est configuré, le fichier de post-configuration du réseau Ethernet 2 est ignoré.

## Configuration d'un module d'E/S distribuées Advantys OTB sur le Scrutateur d'E/S Modbus TCP

### Configuration d'un esclave OTB

Pour configurer l'esclave OTB ajouté au **Scrutateur d'E/S Modbus TCP**, procédez comme suit :

Etape	Action
1	<p>Dans l'arborescence <b>Equipements</b>, double-cliquez sur le nœud de l'équipement Advantys OTB. Résultat : la fenêtre de configuration s'affiche.</p> 
2	Dans le champ <b>Adresse IP esclave</b> , entrez l'adresse IP affectée à l'équipement Advantys OTB.
3	<p>Entrez une valeur de <b>Timeout de validité (ms)</b> (1000 par défaut). Cette valeur représente le délai maximum entre une demande du Scrutateur d'E/S Modbus TCP et la réponse de l'esclave. A l'expiration du timeout de validité, les bits de validité associés prennent la valeur 0. Les valeurs des bits de validité peuvent être visualisées dans l'onglet IOScanner Mappage d'E/S (<i>voir page 27</i>) ou via le serveur Web. Le timeout de validité s'applique à toutes les voies de l'esclave.</p>
4	Entrez une valeur de <b>Vitesse de répétition (ms)</b> (20 par défaut). La valeur de <b>Timeout de validité (ms)</b> doit être supérieure à celle de <b>Vitesse de répétition (ms)</b> .
5	Configurez les E/S de l'équipement Advantys OTB dans l'onglet <b>Configuration des E/S OTB</b> .
6	Ajoutez et configurez les modules d'extension TM2 connectés à l'équipement OTB.
7	Appelez un bloc fonction ( <i>voir page 62</i> ) <code>CONFIGURE_OTB</code> pour mettre à jour la configuration Advantys OTB avec les données entrées aux étapes précédentes.

**NOTE** : Les fonctions expertes de l'équipement Advantys OTB, notamment les compteurs, les compteurs rapides et les générateurs d'impulsions, ne sont pas directement exploitables dans le Scrutateur d'E/S Modbus TCP.

### Compatibilité des modules TM3

Les modules TM3 ne sont pas compatibles avec Advantys OTB.

## Compatibilité des modules TM2

Le tableau suivant répertorie les modules TM2 compatibles avec Advantys OTB :

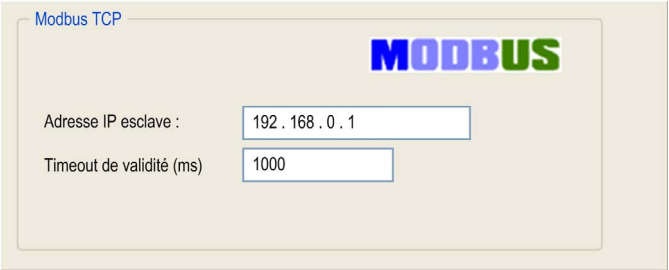
Référence	Type
TM2AMI2HT	2 entrées analogiques
TM2AMI2LT	2 entrées analogiques
TM2AMI4LT	4 entrées analogiques
TM2AMI8HT	8 entrées analogiques
TM2ARI8HT	8 entrées analogiques
TM2ARI8LRJ	8 entrées analogiques
TM2ARI8LT	8 entrées analogiques
TM2AMO1HT	1 sortie analogique
TM2AVO2HT	2 sorties analogiques
TM2AMM3HT	2 entrées analogiques 1 sortie analogique
TM2AMM6HT	4 entrées analogiques 2 sorties analogiques
TM2ALM3LT	2 entrées analogiques 1 sortie analogique
TM2DAI8DT	8 entrées numériques Type de signal : type AC
TM2DDI8DT	8 entrées numériques Type de signal : logique positive/négative
TM2DDI16DT	16 entrées numériques Type de signal : logique positive/négative
TM2DDI16DK	16 entrées numériques Type de signal : logique positive/négative
TM2DDI32DK	32 entrées numériques Type de signal : logique positive/négative
TM2DRA8RT	8 contacts sur 1 ligne commune Type de sortie : relais (contacts NO)
TM2DRA16RT	16 contacts sur 2 lignes communes Type de sortie : relais (contacts NO)
TM2DDO8UT	8 sorties de transistor sur 1 ligne commune Type de signal : logique négative
TM2DDO8TT	8 sorties de transistor sur 1 ligne commune Type de signal : logique positive
TM2DDO16UK	16 sorties de transistor sur 1 ligne commune Type de signal : logique négative

Référence	Type
TM2DDO16TK	16 sorties de transistor sur 1 ligne commune Type de signal : logique positive
TM2DDO32UK	32 sorties de transistor sur 2 lignes communes Type de signal : logique négative
TM2DDO32TK	32 sorties de transistor sur 2 lignes communes Type de signal : logique positive
TM2DMM8DRT	4 entrées numériques Type de signal : logique positive/négative 1 ligne commune avec 4 contacts Type de sortie : relais (contacts NO)
TM2DMM24DRF	16 entrées numériques Type de signal : logique positive/négative 2 lignes communes avec 8 contacts chacune Type de sortie : relais (contacts NO)

## Configuration d'un esclave prédéfini sur le Scrutateur d'E/S Modbus TCP

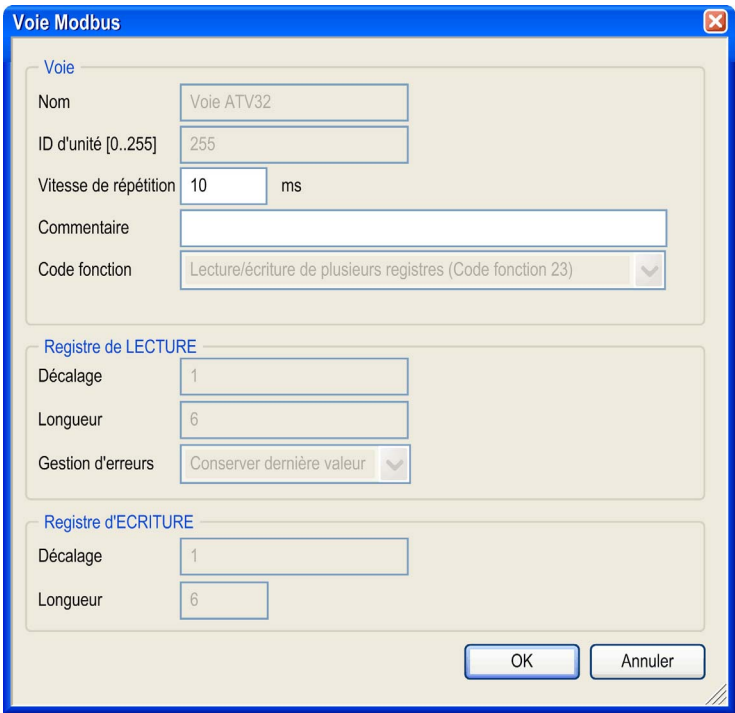
### Configuration d'un esclave prédéfini ajouté au Scrutateur d'E/S Modbus TCP

Pour configurer l'esclave prédéfini ajouté au Scrutateur d'E/S Modbus TCP, procédez comme suit :

Etape	Action
1	<p>Dans l'arborescence <b>Equipements</b>, double-cliquez sur le nœud de l'esclave ajouté. Résultat : la fenêtre de configuration s'affiche.</p> 
2	Dans le champ <b>Adresse IP esclave</b> , entrez l'adresse IP de l'esclave Modbus.
3	<p>Entrez une valeur de <b>Timeout de validité (ms)</b> (1000 par défaut). Cette valeur représente le délai maximum entre une demande du Scrutateur d'E/S Modbus TCP et la réponse de l'esclave. A l'expiration du timeout de validité, les bits de validité associés prennent la valeur 0. Les valeurs des bits de validité peuvent être visualisées dans l'onglet IOScanner Mappage d'E/S (<i>voir page 27</i>) ou via le serveur Web. Le timeout de validité s'applique à toutes les voies de l'esclave.</p>
4	Pour les équipements associés à des paramètres avancés, des paramètres supplémentaires peuvent être requis. Reportez-vous au Guide de l'utilisateur de Device Type Manager.

## Modification de la voie Modbus TCP

Pour modifier les paramètres de voie Modbus pour un esclave prédéfini, procédez comme suit :

Etape	Action
1	Dans l'arborescence <b>Equipements</b> , double-cliquez sur le nœud de l'esclave ajouté.
2	Sélectionnez l'onglet <b>Configuration de voie Modbus TCP</b> et cliquez sur le bouton <b>Modifier...</b> Résultat : la fenêtre <b>Voie Modbus</b> s'affiche.
	
3	Entrez la <b>Vitesse de répétition</b> de la voie. Il s'agit de la fréquence d'interrogation des requêtes Modbus.
4	Vous pouvez également entrer un <b>Commentaire</b> sur la voie.
5	Cliquez sur <b>OK</b> .

## Configuration d'un équipement générique sur le Scrutateur d'E/S Modbus TCP

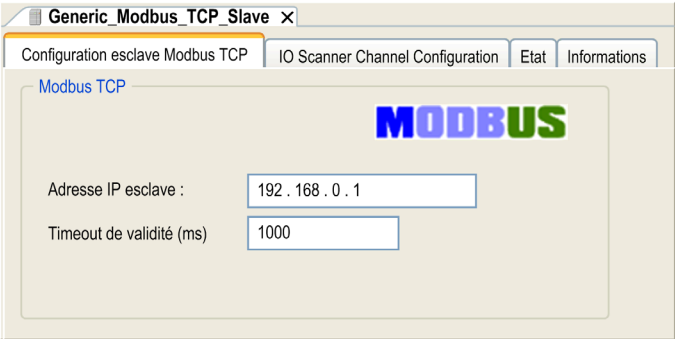
### Présentation

Pour configurer un équipement générique ajouté au Scrutateur d'E/S Modbus TCP, renseignez les paramètres des deux onglets suivants :

- **Configuration esclave Modbus TCP**
- **IO Scanner Channel Configuration**


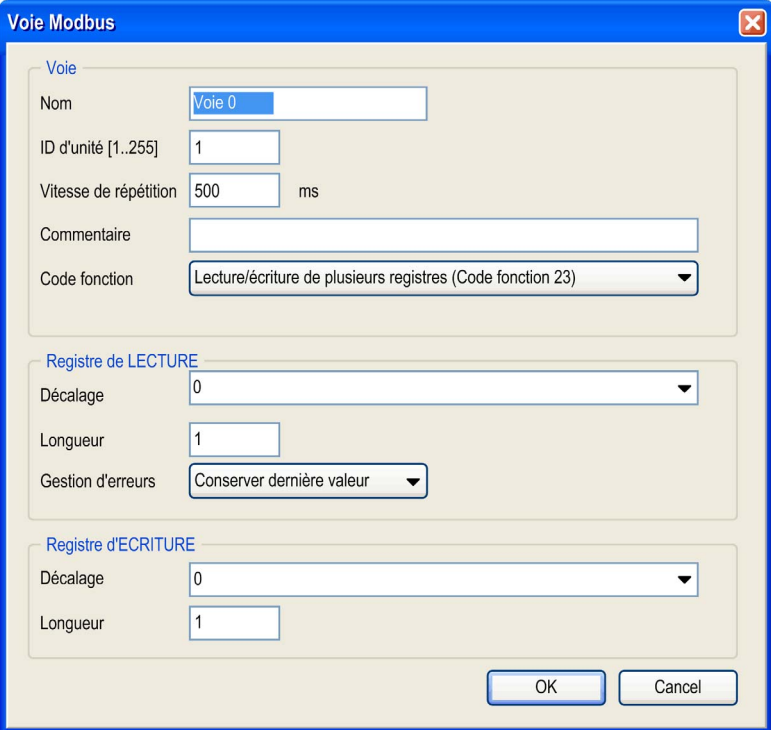
### Onglet Configuration esclave Modbus TCP

Pour configurer les paramètres dans l'onglet **Configuration esclave Modbus TCP**, procédez comme suit :

Etape	Action
1	<p>Dans l'arborescence <b>Equipements</b>, double-cliquez sur <b>Generic_Modbus_TCP_Slave</b>. Résultat : la fenêtre de configuration s'affiche.</p> 
2	<p>Dans le champ <b>Adresse IP esclave</b>, indiquez une valeur (par défaut, 192.168.0.1).</p>
3	<p>Entrez une valeur de <b>Timeout de validité (ms)</b> (1000 par défaut). Cette valeur représente le délai maximum entre une demande du Scrutateur d'E/S Modbus TCP et la réponse de l'esclave. A l'expiration du timeout de validité, les bits de validité associés prennent la valeur 0. Les valeurs des bits de validité sont consultables dans l'onglet IOScanner Mappage E/S (<a href="#">voir page 27</a>).</p>

## Onglet IO Scanner Channel Configuration

Pour configurer les paramètres dans l'onglet **IO Scanner Channel Configuration**, procédez comme suit :

Etape	Action
1	Cliquez sur l'onglet <b>IO Scanner Channel Configuration</b> : 
2	Pour supprimer une voie, sélectionnez-la et cliquez sur <b>Supprimer</b> .
3	Pour modifier les paramètres d'une voie, sélectionnez-la et cliquez sur <b>Modifier</b> .
4	Pour ajouter une voie, cliquez sur <b>Ajouter une voie</b> . La boîte de dialogue suivante s'affiche : 

Etape	Action
5	<p>Dans la zone <b>Voie</b>, vous pouvez définir les paramètres suivants :</p> <ul style="list-style-type: none"> <li>● <b>Nom</b> : nom facultatif de la voie.</li> <li>● <b>ID unité [1..255]</b> : ID d'unité de l'équipement esclave Modbus TCP (par défaut, 255). Voir la remarque.</li> <li>● <b>Vitesse de répétition</b> : intervalle d'interrogation de la requête Modbus (20 ms par défaut)</li> <li>● <b>Commentaire</b> : champ facultatif permettant de fournir une description de la voie</li> <li>● <b>Code de fonction</b> : Type de requête Modbus : <ul style="list-style-type: none"> <li>● <b>Read/Write Multiple Registers (Function code 23)</b> (par défaut)</li> <li>● <b>Read Holding Registers (Function code 03)</b></li> <li>● <b>Write Multiple Registers (Function code 16)</b></li> </ul> </li> </ul> <p>Dans la zone <b>Registre de LECTURE</b>, vous pouvez définir les paramètres suivants :</p> <ul style="list-style-type: none"> <li>● <b>Décalage</b> : numéro du premier registre à lire, de 0 à 65535</li> <li>● <b>Longueur</b> : nombre de registres à lire (selon le code de fonction).</li> <li>● <b>Traitement des erreurs</b> : définissez la valeur de repli en cas d'une interruption de communication : <ul style="list-style-type: none"> <li>● <b>Conserver la dernière valeur</b> : (par défaut) maintient la dernière valeur valide.</li> <li>● <b>Régler sur zéro</b> : ramène toutes les valeurs à 0.</li> </ul> </li> </ul> <p>Dans la zone <b>Registre d'ECRITURE</b>, vous pouvez définir les paramètres suivants :</p> <ul style="list-style-type: none"> <li>● <b>Décalage</b> : numéro du premier registre à écrire, de 0 à 65535</li> <li>● <b>Longueur</b> : nombre de registres à écrire (selon le code de fonction).</li> </ul>
6	Cliquez sur <b>OK</b> pour valider la configuration de la voie.
7	Répétez les étapes 4 à 6 pour créer les autres voies définissant la communication Modbus avec l'équipement. Vous devez créer une voie pour chaque requête Modbus.

**NOTE** : L'identificateur d'unité est utilisé avec les équipements Modbus TCP qui sont composés de plusieurs équipements Modbus, par exemple sur les passerelles de Modbus TCP vers Modbus RTU. Dans ce cas, l'identificateur d'unité permet d'atteindre l'adresse esclave de l'équipement situé derrière la passerelle. Par défaut, les équipements compatibles Modbus/TCP ignorent ce paramètre.

---

# Chapitre 4

## Scrutateur d'E/S Modbus TCP - Fonctionnement

---

### Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
Scrutateur d'E/S Modbus TCP - Vérification des ressources	38
Scrutateur d'E/S Modbus TCP - Modes de fonctionnement	40
Interface d'application	44

## Scrutateur d'E/S Modbus TCP - Vérification des ressources

### Objectif

L'onglet **Ressources du Scrutateur d'E/S Modbus TCP** permet d'estimer la charge pesant sur la fonctionnalité Scrutateur d'E/S Modbus TCP. Vérifiez cette charge avant de mettre la machine en route.

Pour gérer la charge, vous pouvez manipuler les facteurs suivants :

- nombre d'esclaves
- nombre de voies
- vitesse de répétition

### Estimation de la charge

L'équation suivante permet d'estimer la charge imposée au composant Scrutateur d'E/S Modbus TCP :

$$\text{Charge du Scrutateur d'E/S (\%)} = \sum_{\text{voie} = 1}^{\text{Nb de voies}} \frac{50}{\text{Vitesse de répétition voie}}$$

Le logiciel SoMachine fournit un calcul automatique de la charge :

Etape	Action
1	Dans l'arborescence <b>Equipements</b> , double-cliquez sur le noeud <b>Modbus_TCP_IOScanner</b> .
2	Sélectionnez l'onglet <b>Ressources du Scrutateur d'E/S Modbus TCP</b> .
3	Cliquez sur <b>Calculer</b> .

## Description

L'illustration suivante présente l'onglet **Ressources du Scrutateur d'E/S Modbus TCP** :

The screenshot shows the 'Modbus\_TCP\_IOScanner' application window. The 'Ressources du Scrutateur d'E/S Modbus TCP' tab is active. The interface contains the following elements:

Paramètre	Valeur	Indicateur visuel
Nombre de voies configurées	6	2 barres bleues
Mots d'entrée	22	1 barre bleue
Mots de sortie	22	1 barre bleue
Charge du Scrutateur d'E/S (%)	18%	4 barres bleues

A 'Calculer' button is positioned at the bottom left of the main content area.

**NOTE** : La charge doit être inférieure à 100 %.

## Scrutateur d'E/S Modbus TCP - Modes de fonctionnement

### Etats du Scrutateur d'E/S Modbus TCP

Les états du Scrutateur d'E/S Modbus TCP définissent le comportement des différents équipements du réseau Scrutateur d'E/S Modbus TCP. Pour chaque état, les informations de surveillance (bit de validité, états de la communication, etc.) sont spécifiques.

L'état du Scrutateur d'E/S Modbus TCP dépend de l'état du contrôleur :

Etat du contrôleur	Etat du Scrutateur d'E/S Modbus TCP
EMPTY	IDLE
STOPPED	STOPPED
HALT	STOPPED
RUNNING	OPERATIONAL
RUNNING with breakpoint	OPERATIONAL avec un comportement spécifique

### Etat EMPTY du contrôleur

Les connexions TCP/IP sont fermées.

Les états des équipements esclaves sont gérés en fonction de leur mode de fonctionnement individuel. Dans l'état EMPTY, le Scrutateur d'E/S Modbus TCP n'est pas créé. Il n'y a donc ni bits de validité ni images d'E/S disponibles.

L'illustration suivante présente la page du serveur Web pour cet état :

The screenshot shows the web interface for the TM251MESE device. The top navigation bar includes 'Accueil', 'Monitoring', 'Diagnostics', and 'Maintenance'. The 'Diagnostics' menu is open, showing options for 'Controller', 'TM3 Expansion', 'Ethernet', 'Serial', and 'Scanner Status'. The 'Scanner Status' page is displayed, showing the following information:

- Scanner Status:** Idle (indicated by a minus sign icon).
- Statistiques de connexion:**
  - Total transmissions sent: 0
  - Number of Configured Connections: 0
  - IOScanner Execution Time(us): 0
- Scanned Device Statuses:** No Scanned Devices Reported.
- Legend:**
  - Not Configured
  - Scanned
  - Fault

### Etat STOPPED du contrôleur

Les connexions TCP/IP sont fermées. Lorsque le Scrutateur d'E/S Modbus TCP passe de l'état OPERATIONAL à l'état STOPPED, toutes les connexions avec les esclaves sont fermées en mode unilatéral.

Les équipements esclaves sont gérés en fonction de leur mode de fonctionnement individuel.

Le tableau suivant présente les variables SoMachine :

Variable	Valeur	Commentaires
Valeur du bit de validité	0	-
Image d'entrée	0 ou dernière valeur lue	Les valeurs d'entrée dépendent du paramètre <b>Traitement des erreurs</b> . Les valeurs d'entrée correspondent à la situation où le contrôleur est passé en état STOPPED ne reflète donc pas forcément l'état réel de l'entrée depuis lors.

L'illustration suivante présente la page du serveur Web pour cet état :

The screenshot shows the web interface for TM251MESE. The 'Diagnostics' tab is active. The 'I/O Scanner' section displays the following information:

- Scanner Status:** Stopped (indicated by a green checkmark icon).
- Connection Statistics:**
  - Total transmissions sent: 0
  - Number of Configured Connections: 3
  - IOScanner Execution Time(us): 24
- Scanned Device Statuses:** A grid of 64 device status indicators (rows 0-15, 16-31, 32-47, 48-63). The first three devices (0, 1, 2) are marked as 'Not Configured' (grey square with a diagonal line). All other devices are marked as 'Scanned' (green checkmark icon).

Legend for Scanned Device Statuses:

- Not Configured
- Scanned
- Fault

### Etat HALT du contrôleur

Même comportement que pour l'état STOPPED du contrôleur.



**Contrôleur en état RUNNING avec point d'arrêt**

Les connexions TCP/IP sont ouvertes.

Les équipements esclaves sont gérés par le contrôleur.

Le tableau suivant présente les variables SoMachine :

Variable	Valeur	Commentaires
Valeur du bit de validité	1	-
Image d'entrée	Dernière valeur lue	Les valeurs d'entrée correspondent à la situation où le contrôleur est passé en état RUNNING avec point d'arrêt et ne reflète donc pas forcément l'état réel de l'entrée depuis lors.

## Interface d'application

### Présentation

L'interface d'application est un ensemble de fonctions et de variables qui permettent la communication entre l'application et le Scrutateur d'E/S Modbus TCP :

- Tâche de cycle de bus
- Variables d'état
- Variables d'image d'E/S
- Blocs fonction

### Tâche de cycle de bus

Le Scrutateur d'E/S Modbus TCP et l'application échangent des données à chaque cycle d'une tâche de l'application.

Le paramètre **Tâche de cycle de bus** permet de sélectionner la tâche d'application qui gère le Scrutateur d'E/S Modbus TCP :

- **Utiliser les paramètres de cycle du bus supérieur** : associe le Scrutateur d'E/S Modbus TCP à la tâche d'application qui gère le contrôleur.
- **MAST** : associe le Scrutateur d'E/S Modbus TCP à la tâche MAST.
- **Autre tâche existante** : vous pouvez créer une tâche et l'associer au Scrutateur d'E/S Modbus TCP.

Pour plus d'informations sur les tâches d'application, reportez-vous au Guide de programmation SoMachine.

### Variables d'état

Il existe deux types de variable d'état :

- **Bits de validité** : variables indiquant l'état de communication des voies. Il y a un bit de validité par voie.
- **Etat global du scrutateur** : variables indiquant l'état du Scrutateur d'E/S Modbus TCP.

Le tableau suivant présente les valeurs des bits de validité :

Valeur du bit de validité	Etat de communication de la voie
0	Le timeout de validité a expiré sans qu'une réponse soit reçue.
1	Aucune erreur détectée. La requête et la réponse sont reçues.

### Variables d'image d'E/S

Le Scrutateur d'E/S Modbus TCP collecte et écrit des données sur les équipements esclaves. Ces variables sont appelées images d'E/S.

## Adresses des variables

Chaque variable a sa propre adresse :

Variable	Type	Quantité
Image d'E/S	%IW pour les entrées %QW pour les sorties	Un tableau de mots est créé par voie.
Bit de validité	%IW	Quatre mots consécutifs
Etat global du scrutateur	%IW	Un mot

Le mappage d'E/S (*voir page 49*) organise les variables :

- Vérification des adresses
- Affectation de noms aux variables (mappage)

## Blocs fonction contrôlant le Scrutateur d'E/S Modbus TCP

L'application utilise plusieurs blocs fonction (*voir page 57*) pour communiquer avec le contrôleur et les esclaves :

- CONFIGURE\_OTB
- IOS\_GETSTATE
- IOS\_START
- IOS\_GETHEALTH
- IOS\_STOP

## Blocs fonction utilisés pour contrôler les équipements ATV et Lexium

Utilisez les blocs fonction PLC Open pour contrôler les équipements ATV et Lexium. Vous pouvez accéder à ces blocs fonction dans la bibliothèque Modbus TCP Altivar. Pour plus d'informations, reportez-vous aux documents ATV Modbus TCP Function blocks Library Guide et LXM Modbus TCP Function blocks Library Guide.



---

# Chapitre 5

## Scrutateur d'E/S Modbus TCP - Maintenance

---

### Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
Diagnostics : Mode en ligne de SoMachine	48
Diagnostics : Serveur Web	52
Dépannage	54

## Diagnostique : Mode en ligne de SoMachine



### Présentation

Dans le mode en ligne, vous pouvez surveiller le Scrutateur d'E/S Modbus TCP dans SoMachine à l'aide des méthodes suivantes :

- Icônes de l'arborescence **Equipements**
- Onglet Etat du Scrutateur d'E/S Modbus TCP et des différents équipements esclaves
- Onglet Mappage E/S du Scrutateur d'E/S Modbus TCP
- Onglets Mappage E/S des équipements esclaves
- Onglet Ressources du Scrutateur d'E/S Modbus TCP

### Arborescence des équipements

L'état de la communication entre le Scrutateur d'E/S Modbus TCP et les esclaves est représenté par des icônes dans l'arborescence **Equipements** :

Icône	Signification
	La communication avec l'équipement est OK. <b>NOTE</b> : Scrutateur d'E/S Modbus TCP est toujours présenté avec cette icône.
	Le contrôleur n'arrive pas à communiquer avec l'équipement. <b>NOTE</b> : Lorsque le Scrutateur d'E/S Modbus TCP est en état STOPPED, tous les équipements sont présentés avec cette icône.

## Mappage d'E/S du Scrutateur d'E/S Modbus TCP

Scrutateur d'E/S Mappage E/S		Ressources du Scrutateur d'E/S Modbus TCP		Informations				
Voies								
Variable	Mappage	Canal	Adresse	Type	Defaul...	Valeur actuelle	Prepar...	
Diagnostic								
Etat global		Etat global	%I...	UINT	0	2		
Bits de validité								
Bit de validité...		Bit de validité...	%I...	WORD		63		
Healthbits_OTB1EODM9LP		Bit 0	%IX...	BOOL	FALSE	TRUE		
Healthbits_Altivar32		Bit 1	%IX...	BOOL	FALSE	TRUE		
Healthbits_Lexium32M		Bit 2	%IX...	BOOL	FALSE	TRUE		
Healthbits_Generic_Slave_channel3		Bit 3	%IX...	BOOL	FALSE	TRUE		
Healthbits_Generic_Slave_channel4		Bit 4	%IX...	BOOL	FALSE	TRUE		
Healthbits_Generic_Slave_channel5		Bit 5	%IX...	BOOL	FALSE	TRUE		
		Bit 6	%IX...	BOOL	FALSE	FALSE		
		Bit 7	%IX...	BOOL	FALSE	FALSE		
		Bit 8	%IX...	BOOL	FALSE	FALSE		
		Bit 9	%IX...	BOOL	FALSE	FALSE		
		Bit 10	%IX...	BOOL	FALSE	FALSE		
		Bit 11	%IX...	BOOL	FALSE	FALSE		
		Bit 12	%IX...	BOOL	FALSE	FALSE		
		Bit 13	%IX...	BOOL	FALSE	FALSE		
		Bit 14	%IX...	BOOL	FALSE	FALSE		
		Bit 15	%IX...	BOOL	FALSE	FALSE		
		Bit de validité...	%I...	WORD		0		
		Bit de validité...	%I...	WORD		0		
		Bit de validité...	%I...	WORD		0		

Colonne		Utilisation	Commentaire
Variable	Diagnostic	Attribuer un nom à la variable d'état global du scrutateur.	–
	Bits de validité	Attribuer un nom à chaque bit de validité. Par exemple, nommer un bit de validité d'après le nom de l'équipement associé.	Les bits de validité sont regroupés en 4 sous-dossiers de 16 bits.

Colonne	Utilisation	Commentaire
Adresse	Extraire l'adresse de chaque variable.	Il se peut que les adresses changent lorsque la configuration est modifiée.
Valeur actuelle	Surveiller le réseau Scrutateur d'E/S Modbus TCP.	Pour les valeurs booléennes (bit de validité) : <ul style="list-style-type: none"> <li>● <b>TRUE</b> = 1</li> <li>● <b>FALSE</b> = 0</li> </ul>

## Mapping des esclaves

La figure suivante illustre un exemple d'onglet Mappage E/S pour un équipement esclave Advantys OTB :

Configuration esclave Modbus TCP		Configuration des E/S OTB		Scrutateur d'E/S Modbus Mappage E/S			Status	Informations
Voies								
Variable	Mappage	Voie	Adresse	Type	Valeur par défaut	Valeur actuelle		
<b>Entrées</b>								
iwOTB1EODM9LP_Read_Inputs		Read Inputs	%IW18	WORD		2049		
		Bit 0	%IX3...	BOOL	FALSE	TRUE		
		Bit 1	%IX3...	BOOL	FALSE	FALSE		
		Bit 2	%IX3...	BOOL	FALSE	FALSE		
		Bit 3	%IX3...	BOOL	FALSE	FALSE		
		Bit 4	%IX3...	BOOL	FALSE	FALSE		
		Bit 5	%IX3...	BOOL	FALSE	FALSE		
		Bit 6	%IX3...	BOOL	FALSE	FALSE		
		Bit 7	%IX3...	BOOL	FALSE	FALSE		
		Bit 8	%IX3...	BOOL	FALSE	FALSE		
		Bit 9	%IX3...	BOOL	FALSE	FALSE		
		Bit 10	%IX3...	BOOL	FALSE	FALSE		
		Bit 11	%IX3...	BOOL	FALSE	TRUE		
<b>Sorties</b>								
qwOTB1EODM9LP_Output_commands		Output co...	%QW1...	WORD		255		
		Bit 0	%QX2.0	BOOL	FALSE	TRUE		
		Bit 1	%QX2.1	BOOL	FALSE	TRUE		
		Bit 2	%QX2.2	BOOL	FALSE	TRUE		
		Bit 3	%QX2.3	BOOL	FALSE	TRUE		
		Bit 4	%QX2.4	BOOL	FALSE	TRUE		
		Bit 5	%QX2.5	BOOL	FALSE	TRUE		
		Bit 6	%QX2.6	BOOL	FALSE	TRUE		
		Bit 7	%QX2.7	BOOL	FALSE	TRUE		

Colonne		Utilisation	Commentaire
<b>Variable</b>	<b>Entrées</b>	Attribuer un nom à chaque entrée de l'équipement.	Chaque bit peut également être mappé.
	<b>Sorties</b>	Attribuer un nom à chaque sortie de l'équipement.	
<b>Adresse</b>		Extraire l'adresse de chaque variable.	Il se peut que les adresses changent lorsque la configuration est modifiée.
<b>Valeur actuelle</b>		Suivre la valeur en temps réel des entrées de l'équipement.	Pour les valeurs booléennes (chaque bit) : <ul style="list-style-type: none"><li>● <b>TRUE</b> = 1</li><li>● <b>FALSE</b> = 0</li></ul>

## Diagnostics : Serveur Web

### Page Ethernet

**TM251MESE**

Accueil Monitoring **Diagnostics** Maintenance

Diagnostics  
Controller  
TM3 Expansion  
**Ethernet**  
Serial  
Scanner Status

**Ethernet**

**Remote Ping Service**

Enter IP address to ping from Controller:

**Statistics**

Ethernet 1	Ethernet 2
MAC address 0.80.F4.A.1.17	MAC address 0.80.F4.A.1.16
IP address 85.15.3.51	IP address 95.15.1.51
Subnet mask 255.0.0.0	Subnet mask 255.0.0.0
Gateway address 0.0.0.0	Gateway address 0.0.0.0
Status Link up (1)	Status Link up (1)

Ethernet statistics	Modbus statistics
Opened Top connections 2	Messages transmitted OK 181
Frames transmitted OK 355796031	Messages received OK 181
Frames received OK 362481331	Error messages 0
Buffers transmitted NOK 0	IpMaster connection status Not connected (1)
Buffers received NOK 0	IpMaster timeout event counter 0

Ethernet IP statistics
IO Messages transmitted 0

Le tableau suivant présente les résultats du test ping sur la page **Ethernet** :

Icône	Signification
	Echec du test de communication.
	Le contrôleur n'arrive pas à communiquer avec l'adresse IP définie.



## Dépannage

### Principaux problèmes

Symptôme	Cause possible	Résolution
Scrutateur d'E/S Modbus TCP est représenté avec un triangle rouge dans l'arborescence <b>Equipements</b> .	La configuration n'est pas conforme à la version du contrôleur.	<ul style="list-style-type: none"> <li>Nettoyez la configuration.</li> <li>Régénérez l'installation.</li> <li>Vérifiez que le contrôleur a la dernière version du micrologiciel.</li> </ul>
Un esclave est représenté avec un triangle rouge dans l'arborescence <b>Equipements</b> .	Le contrôleur n'arrive pas à communiquer avec l'équipement.	<ul style="list-style-type: none"> <li>Vérifiez le câblage et l'alimentation de l'esclave.</li> <li>Vérifiez l'adresse IP de l'esclave (à l'aide de Remote Ping Service (<i>voir page 52</i>) sur l'adresse IP de l'équipement esclave).</li> <li>Vérifiez si l'esclave prend en charge la requête de lecture/écriture.</li> <li>Vérifiez si les registres indiqués dans la requête sont pertinents pour cet esclave.</li> <li>Vérifiez que les registres indiqués dans la requête ne sont pas protégés en écriture.</li> <li>Vérifiez que le service FDR (remplacement rapide d'équipement) est correctement configuré à l'intérieur de l'esclave.</li> <li>Vérifiez que le paramètre <b>Adresse IP maître</b> est correctement configuré à l'intérieur de l'esclave.</li> </ul>
Une voie esclave est <b>temporairement</b> représentée en rouge.	Le câblage n'est pas stable.	Vérifiez le câblage.
	La configuration nécessite un réglage.	<ul style="list-style-type: none"> <li>Augmentez la valeur du timeout de validité.</li> <li>Augmentez la vitesse de répétition.</li> </ul>
	La charge est trop importante pour le Scrutateur d'E/S Modbus TCP .	Vérifiez l'onglet <b>Ressources du Scrutateur d'E/S Modbus TCP</b> .
Certains états de l'esclave ne sont pas présentés dans l'application.	La vitesse de répétition est trop faible (la valeur du paramètre est trop élevée).	Diminuez la valeur du paramètre Vitesse de répétition pour les voies associées à cet esclave.
	La tâche de cycle de bus n'est pas assez rapide.	<ul style="list-style-type: none"> <li>Associez une autre tâche au Scrutateur d'E/S Modbus TCP.</li> <li>Diminuez la valeur de cycle de la tâche associée.</li> </ul>



## Contenu de cette annexe

Cette annexe contient les chapitres suivants :

Chapitre	Titre du chapitre	Page
A	Scrutateur d'E/S Modbus TCP - Fonctions	57
B	Scrutateur d'E/S Modbus TCP - Types de données	65
C	Représentation des fonctions et blocs fonction	69



---

# Annexe A

## Scrutateur d'E/S Modbus TCP - Fonctions

---

### Présentation

Ce chapitre décrit les fonctions disponibles dans la bibliothèque `ModbusTCPPIOScanner`.

### Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
IOS_GETSTATE : Lit l'état du Scrutateur d'E/S Modbus TCP	58
IOS_START : Lance le Scrutateur d'E/S Modbus TCP	59
IOS_GETHEALTH : Lit la valeur du bit de validité	60
IOS_STOP : Arrête le Scrutateur d'E/S Modbus TCP	61
CONFIGURE_OTB : Envoie la configuration logicielle d'un équipement Advantys OTB	62

## IOS\_GETSTATE : Lit l'état du Scrutateur d'E/S Modbus TCP

### Description de la fonction

Cette fonction renvoie la valeur correspondant à l'état du Scrutateur d'E/S Modbus TCP.

### Représentation graphique



### Représentation en langage IL et ST

Pour voir la représentation générale en langage IL ou ST, consultez le chapitre Représentation des fonctions et blocs fonction ([voir page 69](#))

### Description des variables d'E/S

Ce tableau décrit la variable de sortie :

Sortie	Type	Commentaire
IOS_GETSTATE	IosStateCodes ( <a href="#">voir page 66</a> )	Valeurs renvoyées : IosStateCodes enum

### Exemple

Voici un exemple d'appel de cette fonction :

```
mystate := IOS_GETSTATE() ; (* 0=NOT CONFIGURED 2=OPERATIONAL or
3=STOPPED. *)
```

## IOS\_START : Lance le Scrutateur d'E/S Modbus TCP

### Description de la fonction

Cette fonction démarre le Scrutateur d'E/S Modbus TCP.

Elle permet de contrôler l'exécution du Scrutateur d'E/S Modbus TCP. Par défaut, le Scrutateur d'E/S Modbus TCP démarre comme l'application.

L'appel de cette fonction attend que le Scrutateur d'E/S Modbus TCP soit physiquement démarré, de sorte qu'il peut durer jusqu'à 5 ms.

Le démarrage d'un Scrutateur d'E/S Modbus TCP déjà démarré n'a aucun effet.

### Représentation graphique



### Représentation en langage IL et ST

Pour voir la représentation générale en langage IL ou ST, consultez le chapitre Représentation des fonctions et blocs fonction (*voir page 69*).

### Description des variables d'E/S

Ce tableau décrit la variable de sortie :

Sortie	Type	Commentaire
IOS_START	UDINT	0 = démarrage réussi Autre valeur = échec du démarrage

### Exemple

Voici un exemple d'appel de cette fonction :

```
rc := IOS_START() ;
IF rc <> 0 THEN (* Abnormal situation to be processed at application level
*)
```

## IOS\_GETHEALTH : Lit la valeur du bit de validité

### Description de la fonction

Cette fonction renvoie la valeur du bit de validité d'une voie spécifique.

### Représentation graphique



### Représentation en langage IL et ST

Pour voir la représentation générale en langage IL ou ST, consultez le chapitre Représentation des fonctions et blocs fonction ([voir page 69](#)).

### Description des variables d'E/S

Le tableau suivant décrit la variable d'entrée :

Entrée	Type	Commentaire
channelID	UINT	ID de voie ( <a href="#">voir page 15</a> ) de la voie à surveiller.

Le tableau suivant décrit la variable de sortie :

Sortie	Type	Commentaire
IOS_GETHEALTH	UINT	0 : Les valeurs d'E/S de la voie ne sont pas mises à jour. 1 : Les valeurs d'E/S de la voie sont mises à jour.

### Exemple

Voici un exemple d'appel de cette fonction :

```
chID:=1 ;
channelHealth := IOS_GETHEALTH(chID) (* Get the health value (1=OK, 0=Not
OK) of the channel number chID. The channel ID is displayed in the
configuration editor of the device *)
```

## IOS\_STOP : Arrête le Scrutateur d'E/S Modbus TCP

### Description de la fonction

Cette fonction arrête le Scrutateur d'E/S Modbus TCP.

Elle permet de contrôler l'exécution du Scrutateur d'E/S Modbus TCP. Par défaut, le Scrutateur d'E/S Modbus TCP s'arrête lorsque le contrôleur est en état `STOPPED`.

Le Scrutateur d'E/S Modbus TCP doit être arrêté lors du premier cycle d'application, car cette fonction est un appel synchrone et cela peut prendre un peu de temps.

L'appel de cette fonction peut durer jusqu'à 5 ms car il attend que le Scrutateur d'E/S Modbus TCP s'arrête physiquement.

Arrêter un Scrutateur d'E/S Modbus TCP déjà arrêté n'a aucun effet.

### Représentation graphique



### Représentation en langage IL et ST

Pour voir la représentation générale en langage IL ou ST, consultez le chapitre Représentation des fonctions et blocs fonction ([voir page 69](#)).

### Description des variables d'E/S

Ce tableau décrit la variable de sortie :

Sortie	Type	Commentaire
IOS_STOP	UDINT	0 = arrêt réussi Autre valeur = échec de l'arrêt

### Exemple

Voici un exemple d'appel de cette fonction :

```
rc := IOS_STOP() ;
IF rc <> 0 THEN (* Abnormal situation to be processed at application level
*)
```

## CONFIGURE\_OTB : Envoie la configuration logicielle d'un équipement Advantys OTB

### Description de la fonction

Cette fonction envoie les données de configuration SoMachine d'un Advantys OTB à l'équipement physique via Modbus TCP.

Elle permet de mettre à jour les données de configuration d'un îlot d'E/S sans logiciel tiers.

Le Scrutateur d'E/S Modbus TCP doit être à l'état STOPPED avant l'appel de cette fonction.

L'exécution de ce bloc est asynchrone. Pour vérifier l'achèvement de la configuration, les indicateurs de sortie *Done*, *Busy*, et *Error* doivent être testés à chaque cycle d'application.

### Représentation graphique



### Représentation en langage IL et ST

Pour voir la représentation générale en langage IL ou ST, consultez le chapitre Représentation des fonctions et blocs fonction ([voir page 69](#)).

### Description des variables d'E/S

Le tableau suivant décrit les variables d'entrée :

Entrée	Type	Commentaire
<i>Execute</i>	BOOL	Entrée d'activation. Lance la configuration sur le front montant.
<i>sAddr</i>	STRING	Adresse IP de l'OTB. Cette chaîne doit être au format 3 { xx . xx . xx . xx }

Le tableau suivant décrit les variables de sortie :

Sortie	Type	Commentaire
Done	BOOL	TRUE lorsque la configuration a réussi.
Busy	BOOL	TRUE lorsque la configuration est en cours.
Error	BOOL	TRUE lorsque la configuration s'est terminée avec une erreur détectée.
ConfError	configurationOTBErrorCodes (voir page 68)	Valeurs renvoyées : configurationOTBErrorCodes
CommError	CommunicationErrorCodes (voir page 67)	Valeurs renvoyées : CommunicationErrorCodes

### Exemple

Voici un exemple d'appel de cette fonction :

```

VAR
(*Bloc fonction pour configurer l'OTB. Nécessité d'arrêter le scrutateur d'E/S avant son exécution*)
configure_OTB1: CONFIGURE_OTB;
(*valeur init différente de 16#00000000 ; IO_start_done=0 quand le démarrage réussit*)
IO_start_done: UDINT := 1000;
(*valeur init différente de 16#FFFFFFFF , IO_start_done=16#FFFFFFFF quand l'arrêt réussit*)
IO_stop_done: UDINT := 1000;
(*Configure_OTB_done= true lorsque la configuration de l'OTB réussit. Il est ensuite possible de
démarrer le scrutateur d'E/S.*)
Configure_OTB_done: BOOL;
myBusy: BOOL;
myError: BOOL;
myConfError: configurationOTBErrorCodes;
myCommError: UINT;
myExecute: BOOL;
END_VAR

(*Arrêter le scrutateur d'E/S avant de configurer l'OTB*)
IF NOT myExecute THEN
IO_stop_done:=IOS_STOP();
END_IF

```

(\*Envoyer les données de configuration à l'OTB, à l'adresse IP 95.15.3.1, quand myExecute a la valeur TRUE \*)

```
configure_OTB1(  
Execute:= myExecute,  
sAddr:='3{95.15.3.1}' ,  
Done=> Configure_OTB_done,  
Busy=> myBusy,  
Error=&gt; myError,  
ConfError=&gt; myConfError,  
CommError=&gt; myCommError);
```

(\*Une fois la configuration de l'OTB achevée, démarrer le scrutateur d'E/S\*)

```
IF Configure_OTB_done THEN  
IO_start_done:=IOS_START();  
END_IF
```

---

# Annexe B

## Scrutateur d'E/S Modbus TCP - Types de données

---

### Présentation

Ce chapitre décrit les types de données de la bibliothèque `ModbusTCPIODeviceScanner`.

### Contenu de ce chapitre

Ce chapitre contient les sujets suivants :

Sujet	Page
<code>IoStateCodes</code> : Valeurs d'état Scrutateur d'E/S Modbus TCP	66
<code>CommunicationErrorCodes</code> : Codes d'erreur détectée	67
<code>configurationOTBErrorCodes</code> : Codes d'erreur détectée dans la configuration OTB	68

## IosStateCodes : Valeurs d'état Scrutateur d'E/S Modbus TCP

### Description du type énumération

Le type de données énumération `IosStateCodes` contient les valeurs suivantes :

Enumérateur	Valeur	Commentaire
<code>IosErr</code>	0	Scrutateur d'E/S Modbus TCP en état d'erreur.
<code>IosIdle</code>	1	Scrutateur d'E/S Modbus TCP en état IDLE. La configuration est vide ou non conforme.
<code>IosOperationnal</code>	2	Scrutateur d'E/S Modbus TCP en état OPERATIONAL.
<code>IosStopped</code>	3	Scrutateur d'E/S Modbus TCP en état STOPPED.

## CommunicationErrorCodes : Codes d'erreur détectée

### Description du type énumération

Le type de données énumération `CommunicationErrorCodes` contient les valeurs suivantes :

Enumérateur	Valeur	Commentaire
<code>CommunicationOK</code>	hex 00	L'échange est correct.
<code>TimedOut</code>	hex 01	Arrêt de l'échange pour cause de délai dépassé.
<code>Canceled</code>	hex 02	Arrêt de l'échange sur demande utilisateur.
<code>BadAddress</code>	hex 03	Format d'adresse incorrect.
<code>BadRemoteAddr</code>	hex 04	Adresse distante incorrecte.
<code>BadMgtTable</code>	hex 05	Format de la table de gestion incorrect.
<code>BadParameters</code>	hex 06	Paramètres spécifiques incorrects.
<code>ProblemSendingRq</code>	hex 07	Erreur détectée lors de l'envoi de la requête à sa destination.
<code>RecvBufferTooSmall</code>	hex 09	Taille du tampon de réception insuffisante.
<code>SendBufferTooSmall</code>	hex 0A	Taille du tampon de transmission insuffisante.
<code>SystemResourceMissing</code>	hex 0B	Ressource système manquante.
<code>BadTransactionNb</code>	hex 0C	Numéro de transaction incorrect.
<code>BadLength</code>	hex 0E	Longueur incorrecte.
<code>ProtocolSpecificError</code>	hex FE	L'erreur détectée contient du code spécifique à un protocole.
<code>Refused</code>	hex FF	Transaction refusée.

## configurationOTBErrorCodes : Codes d'erreur détectée dans la configuration OTB

### Description du type énumération

Le type de données énumération `configurationOTBErrorCodes` contient les valeurs suivantes :

Enumérateur	Valeur	Commentaire
<code>ConfigurationOK</code>	hex 00	La configuration OTB est correcte.
<code>IPAddrErr</code>	hex 01	Paramètre d'entrée <code>sAddr</code> incorrect.
<code>ChannelNbErr</code>	hex 02	Absence de valeur d'initialisation de voie OTB pour cette adresse IP.
<code>ChannelInitValueErr</code>	hex 03	Impossible d'obtenir la valeur d'initialisation de voie OTB.
<code>CommunicationErr</code>	hex 04	Arrêt de la configuration OTB suite à une erreur détectée.
<code>IosStateErr</code>	hex 05	Scrutateur d'E/S Modbus TCP en cours d'exécution. Il convient d'arrêter le Scrutateur d'E/S Modbus TCP avant d'exécuter le bloc fonction <code>CONFIGURE_OTB</code> .

---

# Annexe C

## 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	70
Utilisation d'une fonction ou d'un bloc fonction en langage IL	71
Utilisation d'une fonction ou d'un bloc fonction en langage ST	75

## 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.

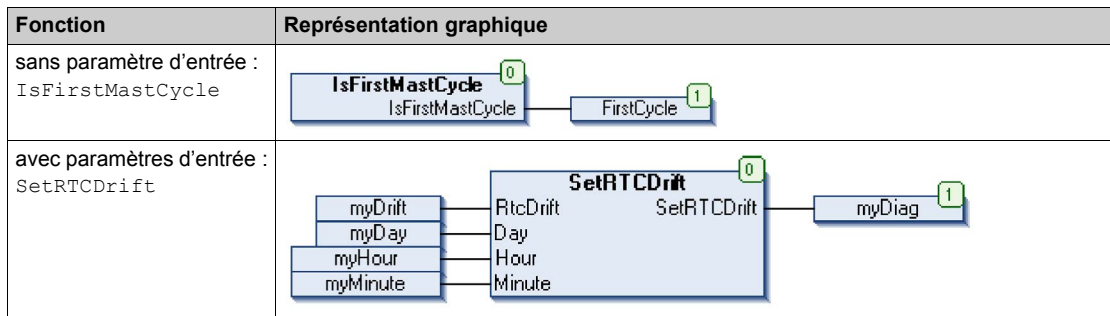
Les fonctions `IsFirstMastCycle` et `SetRTCDrift`, ainsi que le bloc fonction `TON`, sont utilisés à titre d'exemple pour illustrer les mises en œuvre.

### Utilisation d'une fonction en langage IL

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

Etape	Action
1	Ouvrez ou créez un POU en langage IL (Instruction List, ou liste d'instructions). <b>NOTE</b> : La procédure de création d'un POU n'est pas détaillée ici. Pour plus d'informations, reportez-vous à la section Ajout et appel de POU ( <i>voir SoMachine, 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"> <li>● saisissez le nom de la fonction dans la colonne de l'opérateur (champ de gauche), ou</li> <li>● utilisez l'<b>Aide à la saisie</b> pour choisir la fonction (sélectionnez <b>Insérer l'appel de module</b> dans le menu contextuel).</li> </ul>
5	Si la fonction a plusieurs entrées et que l'Aide à la saisie est utilisée, le nombre requis de lignes est automatiquement créé avec ??? dans les champs situés à droite. Remplacez les ??? par la valeur ou la variable appropriée en fonction 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 un nom de variable dans le champ situé à 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 :

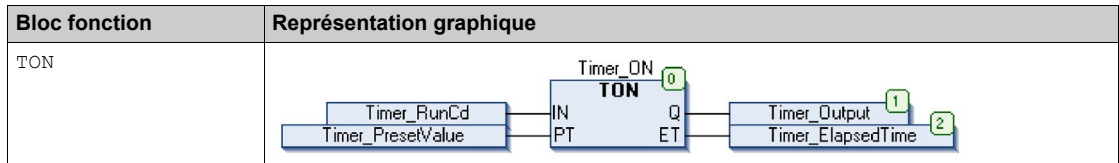
Fonction	Représentation dans l'éditeur IL de POU de SoMachine															
Exemple IL d'une fonction sans paramètre d'entrée : <code>IsFirstMastCycle</code>	<pre> 1  PROGRAM MyProgram_IL 2  VAR 3      FirstCycle: BOOL; 4  END_VAR 5 </pre> <hr/> <table border="1"> <tr> <td>1</td> <td><b>IsFirstMastCycle</b></td> <td></td> </tr> <tr> <td></td> <td>ST</td> <td>FirstCycle</td> </tr> </table>	1	<b>IsFirstMastCycle</b>			ST	FirstCycle									
1	<b>IsFirstMastCycle</b>															
	ST	FirstCycle														
Exemple IL d'une fonction avec des paramètres d'entrée : <code>SetRTCDrift</code>	<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 9 </pre> <hr/> <table border="1"> <tr> <td>1</td> <td><b>LD</b></td> <td>myDrift</td> </tr> <tr> <td></td> <td><b>SetRTCDrift</b></td> <td>myDay</td> </tr> <tr> <td></td> <td></td> <td>myHour</td> </tr> <tr> <td></td> <td></td> <td>myMinute</td> </tr> <tr> <td></td> <td><b>ST</b></td> <td>myDiag</td> </tr> </table>	1	<b>LD</b>	myDrift		<b>SetRTCDrift</b>	myDay			myHour			myMinute		<b>ST</b>	myDiag
1	<b>LD</b>	myDrift														
	<b>SetRTCDrift</b>	myDay														
		myHour														
		myMinute														
	<b>ST</b>	myDiag														

## Utilisation d'un bloc fonction en langage IL

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

Etape	Action
1	Ouvrez ou créez un POU en langage IL (Instruction List, ou liste d'instructions). <b>NOTE</b> : La procédure de création d'un POU n'est pas détaillée ici. Pour plus d'informations, reportez-vous à la section Ajout et appel de POU (voir <i>SoMachine, 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 <code>CAL</code> : <ul style="list-style-type: none"> <li>● Utilisez l'<b>Aide à la saisie</b> pour sélectionner le bloc fonction (cliquez avec le bouton droit et sélectionnez <b>Insérer l'appel de module</b> dans le menu contextuel).</li> <li>● L'instruction <code>CAL</code> et les E/S nécessaires sont automatiquement créées.</li> </ul> Chaque paramètre (E/S) est une instruction : <ul style="list-style-type: none"> <li>● Les valeurs des entrées sont définies à l'aide de « := ».</li> <li>● Les valeurs des sorties sont définies à l'aide de « =&gt; ».</li> </ul>
4	Dans le champ <code>CAL</code> 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 de SoMachine
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</pre> <hr/> <pre>1 CAL Timer_ON(    IN:= Timer_RunCd,    PT:= Timer_PresetValue,    Q=&gt; Timer_Output,    ET=&gt; Timer_ElapsedTime)</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.

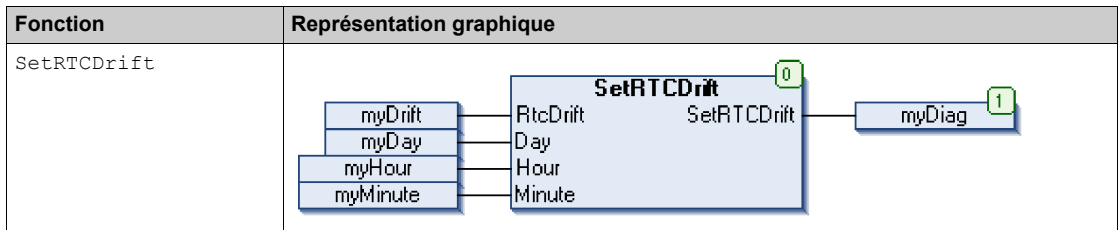
La fonction `SetRTCDrift` et le bloc fonction `TON` sont utilisés à titre d'exemple pour illustrer les mises en œuvre.

### Utilisation d'une fonction en langage ST

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

Etape	Action
1	Ouvrez ou créez un POU en langage ST (Structured Text ou Littéral structuré). <b>NOTE :</b> La procédure de création d'un POU n'est pas détaillée ici. Pour plus d'informations, reportez-vous à la section Ajout et appel de POU ( <i>voir SoMachine, Guide de programmation</i> ).
2	Créez les variables nécessaires à la fonction.
3	Utilisez la syntaxe générale dans l' <b>éditeur ST de POU</b> 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 :

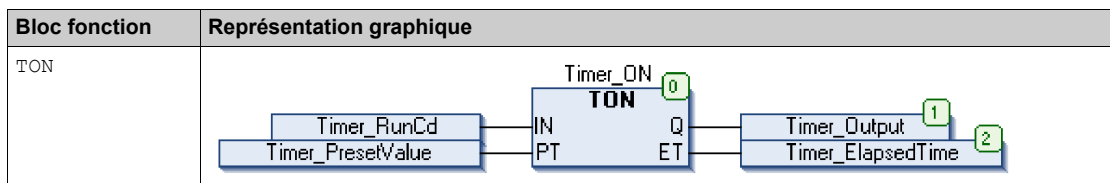
Fonction	Représentation dans l'éditeur ST de POU de SoMachine
SetRTCDrift	<pre>PROGRAM MyProgram_ST VAR myDrift: SINT(-29..29) := 5; myDay: DAY_OF_WEEK := SUNDAY; myHour: HOUR := 12; myMinute: MINUTE; myRTCAdjust: RTCDRIFT_ERROR; END_VAR myRTCAdjust:= SetRTCDrift(myDrift, myDay, myHour, myMinute);</pre>

### Utilisation d'un bloc fonction en langage ST

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

Etape	Action
1	Ouvrez ou créez un POU en langage IL (Instruction List, ou liste d'instructions). <b>NOTE</b> : La procédure de création d'un 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 SoMachine, 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"> <li>• Les variables d'entrée sont les paramètres d'entrée requis par le bloc fonction.</li> <li>• Les variables de sortie reçoivent la valeur renvoyée par le bloc fonction.</li> </ul>
3	Utilisez la syntaxe générale dans l' <b>éditeur ST de POU</b> pour la représentation en langage ST d'un bloc fonction. La syntaxe générale est la suivante : BlocFonction_NomInstance (Entrée1:=VarEntrée1, Entrée2:=VarEntrée2,... Sortie1=>VarSortie1, Sortie2=>VarSortie2,...);

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 de SoMachine
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=&gt;Timer_Output, 5      ET=&gt;Timer_ElapsedTime); </pre>





## B

### bit de validité

Variable qui indique l'état de communication des voies.

## 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.

## D

### DHCP

Acronyme de *dynamic host configuration protocol*. Extension avancée du protocole BOOTP. Bien que DHCP soit plus avancé, DHCP et BOOTP sont tous les deux courants. (DHCP peut gérer les requêtes de clients BOOTP.)

## 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.

### FDR

Acronyme de *fast device replacement*, remplacement rapide d'équipement.

## 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 16#00 à 16#FF en représentation hexadécimale.

## 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****taux de répétition**

Intervalle d'interrogation de la requête Modbus envoyée.

**timeout de validité**

Représente le temps maximum (en ms) entre une requête du scrutateur d'E/S Modbus et une réponse de l'esclave.

**V****variable**

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

**voie Modbus**

Navette de communication qui véhicule une requête Modbus entre le maître et un esclave.





## A

- adresses IP
  - Scrutateur d'E/S Modbus TCP, 20
- architecture
  - Scrutateur d'E/S Modbus TCP, 12

## B

- blocs fonction
  - Scrutateur d'E/S Modbus TCP, 45

## C

- calcul de la charge
  - Scrutateur d'E/S Modbus TCP, 38
- CommunicationErrorCodes
  - types de données, 67
- configurationOTBErrorCodes
  - types de données, 68
- CONFIGURE\_OTB
  - fonctions, 62

## D

- dépannage
  - Scrutateur d'E/S Modbus TCP, 54

## E

- états
  - Scrutateur d'E/S Modbus TCP, 40

## F

- fonctions
  - CONFIGURE\_OTB, 62
  - différences entre une fonction et un bloc

- fonction, 70
- IOS\_GETHEALTH, 60
- IOS\_GETSTATE, 58
- IOS\_START, 59
- IOS\_STOP, 61
- utilisation d'une fonction ou d'un bloc
  - fonction en langage IL, 71
  - utilisation d'une fonction ou d'un bloc
    - fonction en langage ST, 75

## I

- interface d'application
  - Scrutateur d'E/S Modbus TCP, 44
- IOS\_GETHEALTH
  - fonctions, 60
- IOS\_GETSTATE
  - fonctions, 58
- IOS\_START
  - fonctions, 59
- IOS\_STOP
  - fonctions, 61
- iosStateCodes
  - types de données, 66

## M

- Modbus TCP IOScanner
  - adresses IP, 20
- modes de fonctionnement
  - Scrutateur d'E/S Modbus TCP, 40

## P

- planification du réseau
  - Scrutateur d'E/S Modbus TCP, 18
- présentation
  - Scrutateur d'E/S Modbus TCP, 10
- principes
  - Scrutateur d'E/S Modbus TCP, 14

## R

Ressources du Scrutateur d'E/S Modbus  
TCP  
    Scrutateur d'E/S Modbus TCP, 38

## S

scrutateur d'E/S Modbus TCP  
    ajout d'un équipement, 26  
Scrutateur d'E/S Modbus TCP  
    ajout et configuration, 27  
    architecture, 12  
    blocs fonction, 45  
    calcul de la charge, 38  
    configuration d'un équipement générique,  
    34  
    configuration d'un équipement OTB, 29  
    configuration d'un équipement prédéfini,  
    32  
    dépannage, 54  
    états, 40  
    interface d'application, 44  
    modes de fonctionnement, 40  
    onglet Ressources du Scrutateur d'E/S  
    Modbus TCP, 38  
    planification du réseau, 18  
    Présentation, 10  
    principes, 14  
    serveur Web M251, 52  
    tâche de cycle de bus, 44  
    tests du réseau, 23  
Scrutateur d'E/S Modbus TCP  
    surveillance via SoMachine , 48  
serveur Web M251r  
    Scrutateur d'E/S Modbus TCP, 52  
surveillance via SoMachine  
    Scrutateur d'E/S Modbus TCP, 48

## T

tâche de cycle de bus  
    Scrutateur d'E/S Modbus TCP, 44  
tests du réseau  
    Scrutateur d'E/S Modbus TCP, 23

types de données  
    CommunicationErrorCodes, 67  
    configurationOTBErrorCodes, 68  
    IoStateCodes, 66