

JUMO di 308

Indicateur numérique



B 70.1550.2.0

Description de l'interface Modbus



Sommaire

1	Introduction	5
1.1	Préambule	5
1.2	Conventions typographiques	6
1.2.1	Signes d'avertissement	6
1.2.2	Symboles indiquant une remarque	6
1.2.3	Action à mener	7
1.2.4	Types de représentation	7
2	Description du protocole	9
2.1	Principe Maître-Esclave	9
2.2	Mode de transmission (RTU)	9
2.3	Adresse de l'appareil	10
2.4	Déroulement temporel de la transmission	10
2.5	Structure des blocs de données	13
2.6	Différence entre Modbus/J-Bus	13
2.7	Codes de fonction	14
2.7.1	Lecture de n mots	14
2.7.2	Écriture d'un mot	15
2.7.3	Ecriture de n mots	16
2.8	Format de transmission (valeurs entières, flottantes et texte)	17
2.9	Somme de contrôle (CRC16)	19
2.10	Traitement des erreurs	20
3	Interface RS422/485	21
3.1	Schéma de raccordement	21
3.2	Résistance de terminaison	22
3.3	Configuration	23
4	Adresses Modbus	25
4.1	Données de process	25
4.2	Commandes et configuration	27

1.1 Préambule

Cette notice s'adresse aux constructeurs avec formation spécialisée et possédant des connaissances en PC.



Lisez cette notice avant de mettre en service l'interface.
Conservez cette notice dans un endroit accessible à tout moment à tous les utilisateurs.

Aidez-nous à améliorer cette notice en nous faisant part de vos suggestions.

Garantie



Tous les réglages nécessaires sont décrits dans cette notice. Si vous procédez à des manipulations non décrites dans cette notice ou expressément interdites, vous compromettez votre droit à la garantie. En cas de problèmes, veuillez prendre contact avec nos services.

Décharge électrostatique



En cas d'intervention à l'intérieur de l'appareil ou de retour de tiroirs, de blocs ou de composants, il faut respecter les dispositions des normes NF EN 61340-5-1 et NF EN 61340-5-2 « Électrostatique : protection des dispositifs électroniques contre les phénomènes électrostatiques ».

Pour le transport n'utilisez que des emballages **ESD**.

Faites attention aux dégâts provoqués par les décharges électrostatiques, nous dégageons toute responsabilité.

ESD = *Electro Static Discharge* (décharge électrostatique)

1 Introduction

1.2 Conventions typographiques

1.2.1 Signes d'avertissement

Prudence



Ce symbole est utilisé lorsque la non-observation ou l'observation imprécise des instructions peut provoquer des **dommages corporels** !

Attention



Ce symbole est utilisé lorsque la non-observation ou l'observation imprécise des instructions peut **endommager les appareils ou les données** !

ESD



Ce symbole est utilisé lorsqu'il faut prendre ses précautions lors de la manipulation des **composants sensibles aux décharges électrostatiques**.

1.2.2 Symboles indiquant une remarque

Remarque



Ce symbole est utilisé pour attirer votre attention sur un **point particulier**.

Renvoi



Ce symbole renvoie à des **informations complémentaires** dans d'autres notices, chapitres ou sections.

Note de bas de page

abc¹

La note de bas de page est une remarque qui **se rapporte à un endroit précis** du texte. La note se compose de deux parties : le repérage dans le texte et la remarque en bas de page.

Le repérage dans le texte est effectué à l'aide de nombres qui se suivent, mis en exposant.

1.2.3 Action à mener

Instruction *

Ce symbole indique qu'une **action à mener** est décrite. Les différentes étapes sont caractérisées par cette étoile, par ex. :

- * démarrer le logiciel API
- * cliquer sur le catalogue du matériel

Texte à lire absolument



Le texte contient des informations importantes et doit être lu absolument avant de poursuivre.

Chaîne d'instruction

Fichier → Une flèche entre les mots signifie qu'une **série d'instructions** doit être exécutée les unes après les autres.
enregistrer
sous

1.2.4 Types de représentation

Nombre hexadécimal

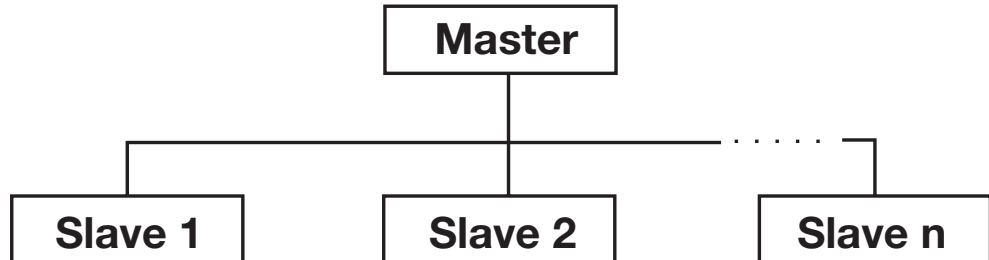
0x0010

Un nombre hexadécimal est précédé de "0x" (ici 16 décimales).

1 Introduction

2.1 Principe Maître-Esclave

La communication entre un appareil maître (par ex. un PC) et un appareil esclave (par ex. système de mesure et de régulation) avec le protocole ModBus a lieu selon le principe maître/esclave sous la forme de demande de données/instruction-réponse.



Le maître contrôle l'échange de données, les esclaves ne donnent que des réponses. Ils sont identifiés par leur adresse d'appareil.

2.2 Mode de transmission (RTU)

Le mode de transmission est le mode RTU (Remote Terminal Unit). La transmission des données s'effectue sous forme binaire (hexadécimale) sur 8 bits. Le bit de poids faible (LSB = least significant bit) est transmis en premier. Le code ASCII n'est pas pris en considération.

Format des données

Le format des données décrit la structure d'un symbole transmis. Les différents formats de données possibles sont les suivants :

Mot de données	Bit de parité	Bit de stop 1/2 bit(s)	Nombre de bits
8 Bit	—	1	9
8 Bit	paire (even)	1	10
8 Bit	impaire (odd)	1	10
8 Bit	—	2	10

2 Description du protocole

2.3 Adresse de l'appareil

L'adresse d'appareil de l'esclave est réglable entre 0 et 254.
L'adresse d'appareil 0 est réservée.



L'interface RS422/RS485 permet d'adresser au maximum 31 esclaves.

Il existe deux possibilités d'échange de données :

Query (consultation)

Demande de données / instruction du maître à un esclave au travers d'une adresse d'appareil particulière.

L'esclave adressé répond.

Broadcast (diffusion)

Instruction du maître à tous les esclaves à l'aide de l'adresse d'appareil 0 (par ex. pour la transmission d'une valeur définie à tous les esclaves).

Les esclaves connectés ne répondent pas. La réception correcte de la valeur par les esclaves devra être contrôlée par une lecture ultérieure de la consigne.

Une demande de données avec l'adresse d'appareil 0 n'est pas logique.

2.4 Déroulement temporel de la transmission

Le début et la fin d'un bloc de données sont caractérisés par des pauses de transmission. Entre deux caractères consécutifs, il doit s'écouler au maximum trois fois le temps de transfert d'un caractère.

Le temps de transfert d'un caractère dépend de la vitesse de transmission (*baudrate*) et du format de données utilisé (nombre de bits de stop et parité).

Pour le format de données 8 bits, sans bit de parité et avec un bit de stop, le temps de transfert d'un caractère est égal à :

Temps de transfert d'un caractère [ms] = 1000 * 9 bits / vitesse de transmission

Pour les autres formats de données :

**Temps de transfert d'un caractère [ms] =
1000 * (8 bits + bit de parité + bit(s) de stop) / vitesse de transmission**

2 Description du protocole

Déroulement

Demande de données du maître Temps de transfert = $n \text{ caractères} * 1000 * x \text{ bits} / \text{vitesse de transmission}$
Identificateur de fin de demande de données $3 \text{ caractères} * 1000 * x \text{ bits} / \text{vitesse de transmission}$
Traitement de la demande de données par l'esclave ($\leq 250 \text{ ms}$)
Réponse de l'esclave Temps de transfert = $n \text{ caractères} * 1000 * x \text{ bits} / \text{vitesse de transmission}$
Identificateur de fin de réponse $3 \text{ caractères} * 1000 * x \text{ bits} / \text{vitesse de transmission}$

Exemple

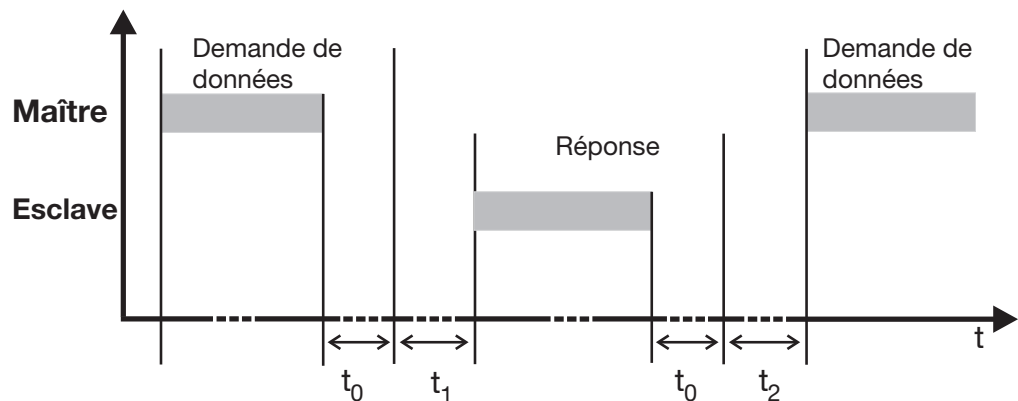
Identificateurs de fin de demande de données et de réponse pour le format 10/9 Bits

Temps d'attente = $3 \text{ caractères} * 1000 * 10 \text{ bits} / \text{vitesse de transmission}$

Vitesse [bauds]	Format de données [bit]	Temps d'attente [ms] (3 caractères)
38400	10	0.79
	9	0.71
19200	10	1.57
	9	1.41
9600	10	3.13
	9	2.82

2 Description du protocole

Chronogramme Une demande de données se déroule selon le chronogramme suivant :



t_0 Identificateur de fin = 3 caractères.
La durée dépend de la vitesse de transmission.

t_1 Cette durée dépend du traitement interne.
La durée maximale de traitement est de 250 ms.



Dans le régulateur, sous le point du menu "Interface", il faut régler le temps de réponse minimal. Il faut laisser s'écouler cette durée avant d'envoyer une réponse (0 à 500 ms). Si on règle une valeur faible, le temps de réponse peut être supérieur à la valeur réglée (le traitement interne dure plus longtemps, le régulateur répond dès que le traitement interne est terminé. Si on règle la valeur sur 0 ms, le régulateur répondra le plus rapidement possible.

Avec une interface RS485, le maître a besoin du temps minimal de réponse (réglable) pour commuter l'interface d'émission en réception. Ce paramètre n'est pas nécessaire pour l'interface RS422.

t_2 L'esclave a besoin de ce temps pour reconfigurer de l'émission en réception. Le maître laisse s'écouler ce temps avant de poser une nouvelle demande de données. Ce temps doit toujours être respecté, même si la nouvelle demande de données est envoyée à un autre appareil.

Interface RS422 : $t_2 = 1\text{ms}$

Interface RS485 : $t_2 = 10\text{ms}$

Le maître ne peut émettre une demande de données, ni à l'intérieur de t_1 et t_2 , ni pendant le temps de réponse. Les demandes pendant t_1 et t_2 sont ignorées par l'esclave. Les demandes pendant le temps de réponse ont pour conséquence que toutes les données qui se trouvent à ce moment sur le bus deviennent invalides.

2 Description du protocole

2.5 Structure des blocs de données

Tous les blocs de données ont la même structure :

Structure des données

Adresse de l'esclave	Code de la fonction	Données	Somme de contrôle CRC16
1 octet	1 octet	x octet	2 octets

Chaque bloc de données dispose de 4 champs :

Adresse de l'esclave	Adresse d'appareil d'un certain esclave
Code de la fonction	Choix de la fonction (lecture, écriture de mots)
Données	Contient les informations : <ul style="list-style-type: none">- adresse des mots- nombre de mots- valeur des mots
Somme de contrôle	Détection des erreurs de transmission

2.6 Différence entre Modbus/J-Bus

Pour certains appareils, une distinction est faite dans la configuration de l'appareil entre Modbus et J-Bus .

Le protocole Modbus est compatible avec le protocole J-Bus, c.-à-d. que la structure des blocs de données est identique. La différence réside uniquement dans le fait que l'adresse du mot (ne pas confondre avec l'adresse de l'esclave) est décalée de 1.

Adresse Modbus	Adresse J-Bus
0x0000	0x0001
0x0001	0x0002
0x0002	0x0003
...	...

Exemple

Adresse pour valeur d'indication Entrée analogique 1 :

Protocole Modbus : 0x0035 (suivant tableau des adresses)

Protocole J-Bus : 0x0036



Les adresses Modbus sont indiquées dans les tableaux d'adresses spécifiques des appareils. Lorsque l'on utilise le protocole J-Bus, les adresses indiquées doivent être incrémentées de 1.

2 Description du protocole

2.7 Codes de fonction

Les fonctions suivantes décrites sont disponibles pour la lecture de valeurs de mesure, de données de process et des appareils ainsi que pour l'écriture de données définies.

Récapitulatif des fonctions

Numéro	Fonction	Limitation
0x03 ou 0x04	Lecture de n mots	max. 32 mots (64 octets)
0x06	Ecriture d'un mot	max. 1 mot (2 octets)
0x10	Ecriture de n mots	max. 32 mots (64 octets)



Lorsque l'appareil ne réagit pas à ces fonctions ou émet un code d'erreur, voir chapitre 2.10 Traitement des erreurs, page 20.

2.7.1 Lecture de n mots

Avec cette fonction n mots ($n \leq 32$) peuvent être lus à partir d'une adresse définie

Demande de données

Adresse de l'esclave	Fonction x03 ou 0x04	Adresse 1er mot	Nombre de mots (max. 32)	Somme de contrôle CRC16
1 octet	1 octet	2 octets	2 octets	2 octets

Réponse

Adresse de l'esclave	Fonction 0x03 ou 0x04	Nombre d'octets lus	Valeur mot(s)	Somme de contrôle CRC16
1 octet	1 octet	1 octet	x octets	2 octets

Exemple

Lecture de la valeur d'indication des entrées analogiques 1 et 2

Adresse du mot = 0x0035 (valeur d'indication de l'entrée analogique 1)

Demande de données :

01	03	0035	0004	5407
----	----	------	------	------

Réponse (valeurs en format Modbus flottant) :

01	03	08	0000	41C8	0000	4120	4A9E
			valeur entrée 1 (25.0)		valeur entrée 2 (10.0)		

2 Description du protocole

2.7.2 Écriture d'un mot

Pour cette fonction, les blocs de données de l'ordre sont identiques aux blocs de données de la réponse.

Ordre

Adresse de l'esclave	Fonction 0x06	Adresse mot	Valeur mot	Somme de contrôle CRC16
1 octet	1 octet	2 octets	2 octets	2 octet

Réponse

Adresse de l'esclave	Fonction 0x06	Adresse mot	Valeur mot	Somme de contrôle CRC16
1 octet	1 octet	2 octets	2 octets	2 octets

Exemple

Écriture de la valeur limite AL seuil d'alarme 1 = 275.0
(valeur = 0x80004389 en format Modbus flottant)

Adresse mot = 0x0043

Ordre : écriture de la première partie de la valeur

01	06	0043	8000	19DE
----	----	------	------	------

Réponse (idem ordre) :

01	06	0043	8000	19DE
----	----	------	------	------

Ordre : écriture de la seconde partie de la valeur

01	06	0044	4389	A81F
----	----	------	------	------

Réponse (idem ordre) :

01	06	0044	4389	A81F
----	----	------	------	------

2 Description du protocole

2.7.3 Ecriture de n mots

Avec cette fonction n mots ($n \leq 32$) peuvent être lus à partir d'une adresse définie.

Ordre

Adresse de l'esclave	Fonction 0x10	Adresse 1er mot	Nombre de mot (max. 32)	Nombre d'octets	Valeur(s) mot	Somme de contrôle CRC16
1 octet	1 octet	2 octets	2 octets	1 octet	x octets	2 octets

Réponse

Adresse de l'esclave	Fonction 0x10	Adresse 1er mot	Nombre de mot	Somme de contrôle CRC16
1 octet	1 octet	2 octets	2 octets	2 octets

Exemple

Ecriture des consignes seuils d'alarme fixes pour LK1 et LK2

Adresse mot = 0x3100 (fenêtre valeur de consigne LK1)

Ordre :

01	10	3100	0004	08	0000	41C8	0000	4120	2A42
					Consigne LK1 (25.0)		Consigne LK2 (10.0)		

Réponse:

01	10	3100	0004	CF36
----	----	------	------	------

2 Description du protocole

2.8 Format de transmission (valeurs entières, flottantes et texte)

Valeurs entières Avec le protocole Modbus, les valeurs entières sont transmises sous la forme suivante : d'abord l'octet de poids fort, ensuite l'octet de poids faible.

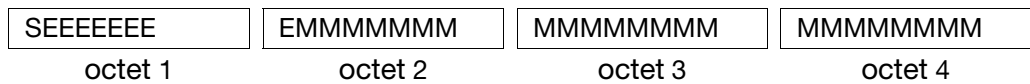
Exemple Consultation de la valeur entière à l'adresse 0x0021 lorsque le contenu à cette adresse est "4" (valeur du mot 0x0004).

Demande : 01 03 0021 0001 (+ 2 octets CRC16)

Réponse : 01 03 02 **0004** (+ 2 octets CRC16)

Valeurs flottantes Le protocole Modbus traite les valeurs flottantes conformément au format standard IEEE 754 (32 bits) ; toutefois il y a une différence : les octets 1 et 2 sont échangés avec les octets 3 et 4.

Format des valeurs flottantes simples (32 bits) suivant le standard 754

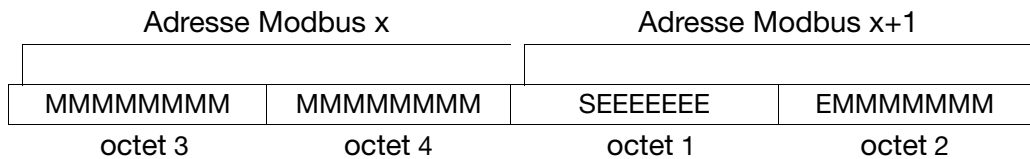


S - Bit de signe

E - Exposant (complément à 2)

M - Mantisse normalisée sur 23 bits

Format des valeurs flottantes avec le protocole Modbus



Exemple Consultation de la valeur flottante à l'adresse 0x0035 lorsque le contenu à cette adresse est 550.0 (0x44098000 au format IEEE 754).

Demande : 01 03 0035 0002 (+ 2 octets CRC16)

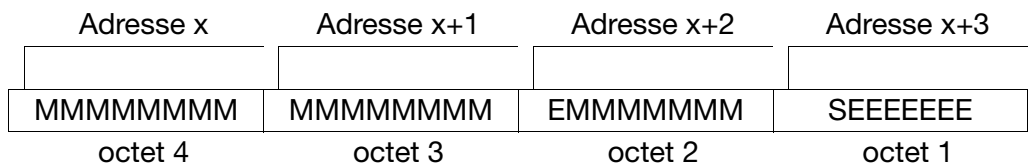
Réponse : 01 03 04 **8000 4409** (+ 2 octets CRC16)

Après le transfert depuis l'appareil, il faut échanger les octets de cette valeur.



De nombreux compilateurs (par ex. Microsoft Visual C++) manipulent les valeurs flottantes dans l'ordre suivant :

Valeur flottante



2 Description du protocole



Déterminez le mode de stockage des valeurs flottantes dans votre application. Le cas échéant, après la consultation de l'enregistreur sans papier, il faudra échanger les octets dans votre programme d'interface.

Chaînes de caractères (textes)

Les chaînes de caractères sont transmises au format ASCII.



Le dernier caractère (indicateur de fin) doit toujours être un "\0" (code ASCII 0x00). Les caractères qui suivent n'ont aucune importance.

Le nombre de caractères max. dans le type de données est indiqué dans les tableaux d'adresse, par ex. "TEXT24" (24 caractères). Si l'on utilise aucun indicateur de fin, il ne reste dans ce cas de figure que 23 caractères lisibles pour le texte.

Si l'on utilise aucun indicateur de fin, il faut, pour l'écriture que le nombre de caractères max. indiqué dans le type de données (par ex. TEXT8 = 8 caractères) soit utilisé. On évite ainsi que des caractères encore en mémoire soient annexés au texte.

Comme la transmission des textes a lieu également mot par mot (16 bits), il faut envoyer un 0x00 supplémentaire si le nombre de caractères est impair (y compris le caractère "\0").

Exemple pour type de données TEXT8

Lecture du nom des grandeurs de process E1 (ici : "INPUT1 ") sous adresse 0x3C80 (8 caractères max. peuvent être mémorisés)

Code ASCII pour "INPUT1 " (avec 2 espaces à la fin) :
0x49, 0x4E, 0x50, 0x55, 0x54, 0x31, 0x20, 0x20

Demande : 01 03 3C80 0004 (+ 2 octets CRC16)

Adresse de l'esclave = 01

Fonction = 03, c.-à-d. lecture de n mots

Adresse = 3C80

Nombre de mots à lire = 0004, max. 8 caractères

Réponse : 01 03 08 **49 4E 50 55 54 31 20 20** (+ 2 octets CRC16)

Adresse de l'esclave = 01

Fonction = 03, c.-à-d. lecture de n mots

Nombre d'octets lus = 08

Variante :

Code ASCII pour "INPUT1" (sans espace à la fin) :

0x49, 0x4E, 0x50, 0x55, 0x54, 0x31, 0x00

ASCII 0x00 ("\0") signifie que la chaîne de caractères s'arrête ici.

Lors de la transmission un **0x00** supplémentaire est annexé, pour obtenir un nombre pair de caractères.

Réponse dans ce cas : 01 03 08 **49 4E 50 55 54 31 00 00** (+ 2 octets CRC16)

2 Description du protocole

2.9 Somme de contrôle (CRC16)

La somme de contrôle (CRC16) permet de détecter les erreurs de transmission. Si une erreur est détectée lors de l'analyse, l'appareil correspondant ne répond pas.

Mode de calcul

CRC = 0xFFFF	
CRC = CRC XOR ByteOfMessage	
For (1 à 8)	
CRC = SHR(CRC)	
if (drapeau report à droite = 1)	
then	else
CRC = CRC XOR 0xA001	
while (tous les octets du message ne sont pas traités);	



L'octet faible de la somme de contrôle est transféré le premier, suivi de l'octet fort.

Exemple

Demande de données : lecture de 2 mots à partir de l'adresse 0x00CE (CRC16 = 0x92A5)

07	03	00	CE	00	02	A5	92
						CRC16	

Réponse : (CRC16 = 0xF5AD)

07	03	04	00	00	41	C8	AD	F5
			mot 1		mot 2		CRC16	

2 Description du protocole

2.10 Traitement des erreurs

Codes d'erreur

Codes d'erreur possibles :

- 1 fonction invalide
- 2 Adresse de paramètres invalide ou nombre de mots ou de bits à lire ou à écrire trop élevé
- 8 Paramètre protégé en écriture

Réponse en cas d'erreur

Adresse esclave	Fonction XX OR 80h	Code d'erreur	Somme de contrôle CRC16
1 octet	1 octet	1 octet	2 octets

Le code de la fonction est associé à 0x80 à l'aide d'une fonction OU (OR), c'est-à-dire que le bit de poids fort (MSB = Most Significant Bit) est mis à 1.

Exemple

Demande de données :

01	03	40	00	00	04	CRC16
----	----	----	----	----	----	-------

Réponse (avec le code d'erreur):

01	83	02	CRC16
----	----	----	-------

Cas particuliers

Dans les cas suivants, l'esclave ne répond pas :

- vitesse de transmission en baud et/ou format de données ne concordent pas pour le maître et pour l'esclave
- L'adresse de l'appareil utilisée ne concorde pas avec l'adresse de l'esclave
- La somme de contrôle (CRC16) est incorrecte.
- L'ordre du maître est incomplet ou contradictoire
- Le nombre de mots à lire est égal à 0

Dans ces cas, la demande de données devrait après écoulement du temps du timeout (2s) être renvoyée.

3.1 Schéma de raccordement

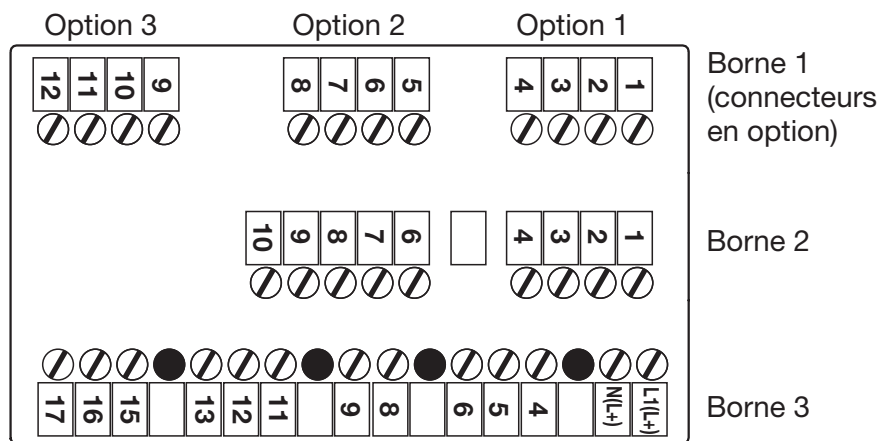
Equipement

L'interface RS422/485 peut être montée à l'un des trois connecteurs en option. Connecteur au choix, seul un module d'interface ne peut être utilisé.



Lorsque l'appareil est équipé en exécution d'usine, la position du module d'interface doit être identifiée à l'aide de l'identification du type. Vous trouverez ces renseignements dans la notice de mise en service B 70.1550.0.

Vue des bornes



Affectation en option de la borne 1

	RS422	RS485
Option 1 1	— RxD +	
2	— RxD -	
3	— TxD +	— RxD/TxD +
4	— TxD -	— RxD/TxD -
Option 2 5	— RxD +	
6	— RxD -	
7	— TxD +	— RxD/TxD +
8	— TxD -	— RxD/TxD -
Option 3 9	— RxD +	
10	— RxD -	
11	— TxD +	— RxD/TxD +
12	— TxD -	— RxD/TxD -

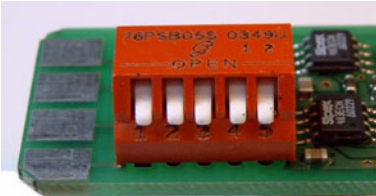
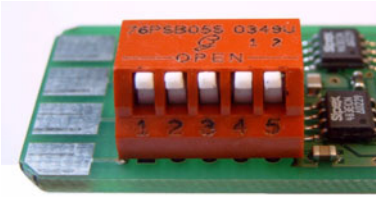
3 Interface RS422/485

3.2 Résistance de terminaison

Réglage


Pour que plusieurs appareils puissent travailler en ligne sans problème, il est nécessaire d'activer leurs résistances de terminaison internes au début et à la fin.

- * Retirer la partie embrochable du boîtier vers l'avant en appuyant sur les surfaces cannelées
- * Placer à l'aide d'un stylo tous les commutateurs blancs dans la même position

Résistance de terminaison du bus active:	* Basculer les 5 commutateurs vers le bas 
Pas de résistance de terminaison (d'usine)	* Basculer les 5 commutateurs vers le haut 

- * Replacer la partie embrochable dans le boîtier

Contrôle

- * Appuyer sur les touches **PGM** + 

Lorsque les résistances de terminaison sont actives, une décimale supplémentaire est représentée derrière le numéro de version (affichage supérieur) durant le contrôle de la version du logiciel.

Résistance active (exemple avec numéro de version 01.01) : 0 1.0 1.

Résistance non active : 0 1.0 1

3.3 Configuration

Les différents réglages de l'interface RS422/485 qui sont effectués au niveau de configuration sont décrits dans le tableau ci-dessous (CONFIG → INTERFCE → RS422485).



Vous trouverez des informations complémentaires pour la configuration dans la notice de mise en service B 70.1550.0.

	Symbole	Valeur/ Choix	Description
Type de protocole	PROTOCOL	0	Modbus
		1	Modbus integer
Vitesse de transmission	BAUDRATE	0	9600 Baud
		1	19200 Baud
		2	38400 Baud
Format des données	FORMATD	0	8 bits de données, 1 bit stop, pas de parité
		1	8 bits de données, 1 bit stop, parité impaire
		2	8 bits de données, 1 bit stop, parité paire
		3	8 bits de données, 2 bits stop, pas de parité
Adresse app.	ADDRESS	0...1...255	Adresses dans le réseau de données
Temps de réponse min.		0...500ms	Laps de temps, entre le moment de la demande d'un appareil dans un réseau de données et la réponse de l'indicateur (réglable uniquement via le logiciel Setup).

Les réglages d'usine sont en **gras**.



L'interface RS422/485 est inactive si communication via l'interface Setup.

3 Interface RS422/485

4 Adresses Modbus

Type de données, type d'accès

Vous trouverez dans le tableau ci-dessous toutes les valeurs de process (variables) avec leurs adresses, le type de données ainsi que le mode d'accès.

Signification :

INT	Integer (8 ou 16 bits)
Bit x	N° bit x
FLOAT	Valeur flottante (4 octets) suivant IEEE 754
TEXT8	Texte 8 caractères
TEXT24	Texte 24 caractères
R/O	Lecture seule
R/W	Ecriture et lecture

4.1 Données de process

Adresses	Type de données / numéro de bit	Accès	Désignation du signal
0x0021	INT	R/O	Sorties binaires 1 à 10 (état de commutation 0 = OFF / 1 = ON)
	Bit 0		Sortie binaire 1 (= 0x0001)
	Bit 1		Sortie binaire 2 (= 0x0002)
	Bit 2		Sortie binaire 3 (= 0x0004)
	Bit 3		Sortie binaire 4 (= 0x0008)
	Bit 4		Sortie binaire 5 (= 0x0010)
	Bit 5		Sortie binaire 6 (= 0x0020)
	Bit 6		Sortie binaire 7 (= 0x0040)
	Bit 7		Sortie binaire 8 (= 0x0080)
	Bit 8		Sortie binaire 9 (= 0x0100)
	Bit 9		Sortie binaire 10 (= 0x0200)
0x0023	INT	R/O	Entrées binaires 1 à 8 (état de commutation 0 = ouvert / 1 = fermé)
	Bit 0		Entrée binaire 1 (= 0x0001)
	Bit 1		Entrée binaire 2 (= 0x0002)
	Bit 2		Entrée binaire 3 (= 0x0004)
	Bit 3		Entrée binaire 4 (= 0x0008)
	Bit 4		Entrée binaire 6 (= 0x0010)
	Bit 5		Entrée binaire 5 (= 0x0020)
	Bit 6		Entrée binaire 8 (= 0x0040)
	Bit 7		Entrée binaire 7 (= 0x0080)

4 Adresses Modbus

Adresses	Type de données / numéro de bit	Accès	Désignation du signal
0x0026	INT	R/O	Signaux binaires (4LK-2LOGIQUES)
	Bit 8		SEUIL D'ALARME 1 (= 0x0100)
	Bit 9		SEUIL D'ALARME 2 (= 0x0200)
	Bit 10		SEUIL D'ALARME 3 (= 0x0400)
	Bit 11		SEUIL D'ALARME 4 (= 0x0800)
	Bit 12		Formule logique 1 (= 0x1000)
	Bit 13		Formule logique 2 (= 0x2000)
0x0027	INT	R/W	Commande Sorties binaires
	Bit 0		Sortie binaire 1 (= 0x8001)
	Bit 1		Sortie binaire 2 (= 0x8002)
	Bit 2		Sortie binaire 3 (= 0x8004)
	Bit 3		Sortie binaire 4 (= 0x8008)
	Bit 4		Sortie binaire 5 (= 0x8010)
	Bit 5		Sortie binaire 6 (= 0x8020)
	Bit 6		Sortie binaire 7 (= 0x8040)
	Bit 7		Sortie binaire 8 (= 0x8080)
	Bit 8		Sortie binaire 9 (= 0x8100)
	Bit 9		Sortie binaire 10 (= 0x8200)
	Bit 15		Activation = 1 (=0x8000)
0x0030	FLOAT	R/O	Pt100 interne [Ohm]
0x0034	INT	R/O	Cadence de scrutation
0x0035	FLOAT	R/O	Entrée analogique 1 [valeur d'indication]
0x0037	FLOAT	R/O	Entrée analogique 2 [valeur d'indication]
0x003D	FLOAT	R/O	Mathématique 1
0x003F	FLOAT	R/O	Mathématique 2
0x3300	FLOAT	R/O	Valeur mini Entrée analogique 1 [valeur d'indication]
0x3302	FLOAT	R/O	Valeur mini Entrée analogique 2 [valeur d'indication]
0x3308	FLOAT	R/O	Valeur maxi Entrée analogique 1 [valeur d'indication]
0x330A	FLOAT	R/O	Valeur maxi Entrée analogique 2 [valeur d'indication]
0x3400	FLOAT	R/O	Valeur Hold Entrée analogique 1 [valeur d'indication]
0x3402	FLOAT	R/O	Valeur Hold Entrée analogique 2 [valeur d'indication]

4.2 Commandes et configuration

Adresses	Type de données / numéro de bit	Accès	Désignation du signal
0x0041	INT	R/W	Fonctions binaires
	Bit 1		Indication valeur Hold (0x0002)
	Bit 2		Validation seuil d'alarme (0x0004)
	Bit 3		Reset valeur min/max. (0x0008)
	Bit 4		Reset fonction tarage (0x0010)
	Bit 5		Fonction tarage (0x0020)
	Bit 6		Validation valeur Hold (0x0040)
0x0042	INT	R/W	Fonctions binaires Affichage/Commande
	Bit 0		Verrouillage du clavier (0x0001)
	Bit 1		Verrouillage configuration (0x0002)
	Bit 4		Extinction de l'écran (touches inactives) (0x0010)
	Bit 5		Changement de la couleur (0x0020)
	Bit 6		Affichage texte (0x0040)
	Bit 7		Saut au prochain paramètre de défilement (0x0080)
0x0043	FLOAT	R/W	Seuil d'alarme 1 valeur limite AL
0x0045	FLOAT	R/W	Seuil d'alarme 1 différentiel de coupure
0x0047	INT	R/W	Seuil d'alarme 1 Enclenchement retardé
0x0048	INT	R/W	Seuil d'alarme 1 Déclenchement retardé
0x0049	FLOAT	R/W	Seuil d'alarme 2 valeur limite AL
0x004B	FLOAT	R/W	Seuil d'alarme 2 différentiel de coupure
0x004D	INT	R/W	Seuil d'alarme 2 Enclenchement retardé
0x004E	INT	R/W	Seuil d'alarme 2 Déclenchement retardé
0x004F	FLOAT	R/W	Seuil d'alarme 3 valeur limite AL
0x0051	FLOAT	R/W	Seuil d'alarme 3 différentiel de coupure
0x0053	INT	R/W	Seuil d'alarme 3 Enclenchement retardé
0x0054	INT	R/W	Seuil d'alarme 3 Déclenchement retardé
0x0055	FLOAT	R/W	Seuil d'alarme 4 valeur limite AL
0x0057	FLOAT	R/W	Seuil d'alarme 4 différentiel de coupure
0x0059	INT	R/W	Seuil d'alarme 4 Enclenchement retardé
0x005A	INT	R/W	Seuil d'alarme 4 Déclenchement retardé
0x005B	FLOAT	R/W	Indicateur analogique
0x005D	INT	R/W	Indicateur binaire
0x3100	FLOAT	R/W	Seuil d'alarme 1 Consigne LK fixe
0x3102	FLOAT	R/W	Seuil d'alarme 2 Consigne LK fixe
0x3104	FLOAT	R/W	Seuil d'alarme 3 Consigne LK fixe
0x3106	FLOAT	R/W	Seuil d'alarme 4 Consigne LK fixe
0x3A00	TEXT24	R/W	Texte d'alarme B1
0x3A0C	TEXT24	R/W	Texte d'alarme B2

4 Adresses Modbus

Adresses	Type de données / numéro de bit	Accès	Désignation du signal
0x3A18	TEXT24	R/W	Texte d'alarme B3
0x3A24	TEXT24	R/W	Texte d'alarme B4
0x3A30	TEXT24	R/W	Texte d'alarme B5
0x3A3C	TEXT24	R/W	Texte d'alarme B6
0x3A48	TEXT24	R/W	Texte d'alarme B7
0x3A54	TEXT24	R/W	Texte d'alarme B8
0x3A60	TEXT24	R/W	Texte d'alarme LK1
0x3A6C	TEXT24	R/W	Texte d'alarme LK2
0x3A78	TEXT24	R/W	Texte d'alarme LK3
0x3A84	TEXT24	R/W	Texte d'alarme LK4
0x3A90	TEXT24	R/W	Texte d'alarme L1
0x3A9C	TEXT24	R/W	Texte d'alarme L2
0x3C00	TEXT24	R/W	Texte process
0x3C80	TEXT8	R/W	Nom grandeur de process E1
0x3C84	TEXT8	R/W	Nom grandeur de process E2
0x3C90	TEXT8	R/W	Nom Mathématique 1
0x3C94	TEXT8	R/W	Nom Mathématique 2
0x3200	FLOAT	R/W	Mathématique 1 (descriptible)
0x3202	FLOAT	R/W	Mathématique 2 (descriptible)
0x3204	FLOAT	R/W	Entrée analogique E1 (descriptible)
0x3206	FLOAT	R/W	Entrée analogique E2 (descriptible)
0x320C	INT	R/W	Logique (descriptible)
	Bit 0		Logique L1 (0x0001)
	Bit 1		Logique L2 (0x0002)
	Bit 6		Activation L2 (0x004X) / Désactivation (0x0000)
	Bit 7		Activation L1 (0x008X) / Désactivation (0x0000)



L'accès direct à la mémoire RAM de l'appareil est possible via le Modbus, afin de lire et d'écrire les valeurs de mesure des entrées analogiques (adresses 0x3204, 0x3206), les résultats des fonctions mathématiques (adresses 0x3200, 0x3202) ainsi que les fonctions logiques (adresse 0x320C).

Une plage comprise entre -19999 et +99999 est disponible pour l'écriture des valeurs de l'entrée analogique et de la fonction mathématique. Est alors utilisée la valeur écrite dans l'appareil au lieu de la valeur originale. Il est ainsi possible par ex. de transférer à l'appareil via le Modbus une valeur mesurée à un autre endroit, pour l'afficher ou l'utiliser comme valeur d'entrée pour d'autres fonctions (seuil d'alarme, mathématique).

Si la valeur originale doit à nouveau être utilisée, il faut écrire la valeur 200000 via le Modbus dans les emplacements mémoires respectifs.

4 Adresses Modbus



Des textes déjà existant dans l'appareil sont perdus à l'écriture de textes via le Modbus. Si de nouveaux textes doivent être écrits par l'intermédiaire du Modbus, je vous recommande d'extraire les textes déjà existants et de les archiver en tant qu'information. Il n'est possible de récupérer les textes initiaux qu'en les retransférant par le logiciel Setup ou le Modbus.



Les opérations d'écriture sur les paramètres R/W provoquent une sauvegarde dans l'EEPROM. Ces modules mémoires n'ont qu'un nombre limité de cycles d'écriture (env. 10000), c'est pourquoi cette fonction peut être déconnectée en cas de programmation fréquente. Les valeurs de paramètres sont alors seulement enregistrées dans la mémoire volatile (RAM) et sont perdues en cas de panne de secteur.

⇒ *Setup/Uniquement Setup/Paramètre non documenté/Paramètre Bit/
Activer paramètre 2*

4 Adresses Modbus



JUMO GmbH & Co. KG

Adresse :

Moltkestraße 13 - 31
36039 Fulda, Allemagne

Adresse de livraison :

Mackenrodtstraße 14
36039 Fulda, Allemagne

Adresse postale :

36035 Fulda, Allemagne

Téléphone : +49 661 6003-0

Télécopieur : +49 661 6003-500

E-Mail : mail@jumo.net

Internet : www.jumo.net

JUMO Régulation SAS

Actipôle Borny

7 Rue des Drapiers

B.P. 45200

57075 Metz - Cedex 3, France

Téléphone : +33 3 87 37 53 00

Télécopieur : +33 3 87 37 89 00

E-Mail : info@jumo.fr

Internet : www.jumo.fr

Service de soutien à la vente :

0892 700 733 (0,337 Euro/min)

JUMO Automation

S.P.R.L. / P.G.M.B.H. / B.V.B.A.

Industriestraße 18

4700 Eupen, Belgique

Téléphone : +32 87 59 53 00

Télécopieur : +32 87 74 02 03

E-Mail : info@jumo.be

Internet : www.jumo.be

JUMO Mess- und Regeltechnik AG

Laubisrütistrasse 70

8712 Stäfa, Suisse

Téléphone : +41 44 928 24 44

Télécopieur : +41 44 928 24 48

E-Mail : info@jumo.ch

Internet : www.jumo.ch