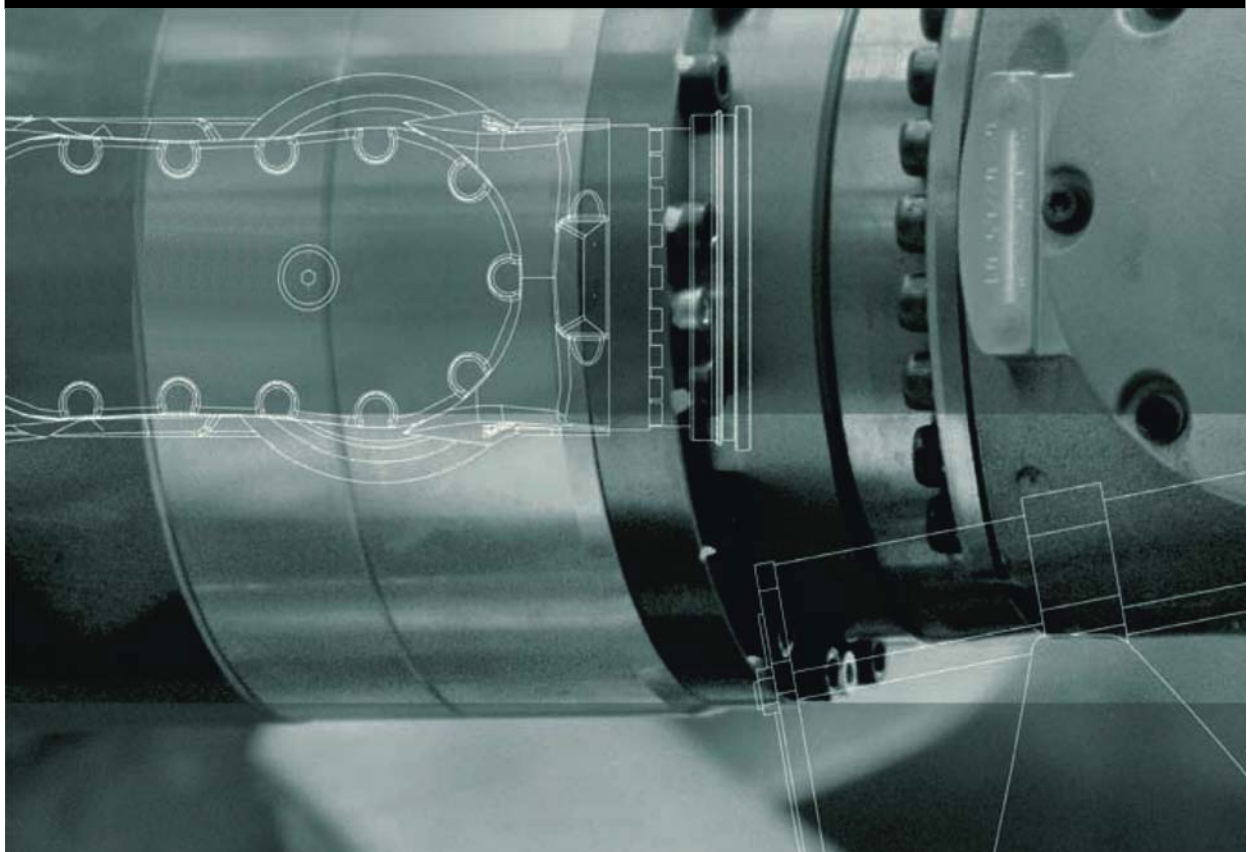


KUKA.RobotSensorInterface 3.3

Pour logiciel KUKA System Software 8.3 et 8.4



Edition: 05.12.2014

Version: KST RSI 3.3 V2

© Copyright 2014

KUKA Roboter GmbH
Zugspitzstraße 140
D-86165 Augsburg
Allemagne

La présente documentation ne pourra être reproduite ou communiquée à des tiers, même par extraits, sans l'autorisation expresse du KUKA Roboter GmbH.

Certaines fonctions qui ne sont pas décrites dans la présente documentation peuvent également tourner sur ce contrôleur. Dans ce cas, l'utilisateur ne pourra exiger ces fonctions en cas de nouvelle livraison ou de service après-vente.

Nous avons vérifié la concordance entre cette brochure et le matériel ainsi que le logiciel décrits. Des différences ne peuvent être exclues. Pour cette raison, nous ne pouvons garantir la concordance exacte. Les informations de cette brochure sont néanmoins vérifiées régulièrement afin d'inclure les corrections indispensables dans l'édition suivante.

Sous réserve de modifications techniques n'influençant pas les fonctions.

Traduction de la documentation originale

KIM-PS5-DOC

Publication:	Pub KST RSI 3.3 (PDF) fr
Structure de livre:	KST RSI 3.3 V2.1
Version:	KST RSI 3.3 V2

Table des matières

1	Introduction	7
1.1	Groupe cible	7
1.2	Documentation du robot industriel	7
1.3	Représentation des remarques	7
1.4	Termes utilisés	8
1.5	Marques	10
2	Description du produit	11
2.1	Aperçu de RobotSensorInterface	11
2.2	Fonctionnement du traitement des signaux	12
2.3	Fonctionnement de la transmission des données	13
2.3.1	Transmission des données via un système E/S	13
2.3.2	Transmission des données via Ethernet	13
2.4	Mode de fonctionnement de la correction de capteur	15
3	Sécurité	21
3.1	Remarques relatives à la sécurité	21
4	Installation	23
4.1	Conditions requises par le système	23
4.2	Installation ou mise à jour de RobotSensorInterface	23
4.3	Désinstallation de RobotSensorInterface	24
4.4	Installation de RSI Visual sur un PC externe	24
4.5	Désinstallation de RSI Visual	25
5	Configuration	27
5.1	Connexion réseau via la KLI du contrôleur du robot	27
5.2	Configuration du réseau de capteurs Ethernet	27
5.3	Modification des variables locales dans le RSI.DAT	27
6	Commande	29
6.1	Aperçu de l'interface utilisateur RSI Visual	29
6.1.1	Ouverture de l'éditeur de flux de signaux	30
6.1.2	Liaison des entrées de signaux d'entrée et de sortie	30
6.1.3	Insertion et liaison d'un commentaire	30
6.1.4	Réglage du paramètre d'objet RSI	31
6.1.5	Activation du paramètre d'objet RSI	31
6.1.6	Sauvegarde de la configuration du flux de signaux	31
6.1.7	Chargement de la configuration du flux de signaux	32
6.2	Aperçu de l'interface utilisateur RSI Monitor	32
6.2.1	Définition des propriétés des signaux	33
6.2.2	Affichage de la forme des signaux	33
6.2.3	Sauvegarde de l'enregistrement des signaux	34
6.2.4	Chargement de l'enregistrement des signaux sur l'écran	34
7	Programmation	35
7.1	Aperçu des instructions RSI	35
7.1.1	Caractères et types	35

7.1.2	RSI_CREATE()	35
7.1.3	RSI_DELETE()	36
7.1.4	RSI_ON()	36
7.1.5	RSI_OFF()	37
7.1.6	RSI_MOVECORR()	38
7.1.7	RSI_GETPUBLICPAR()	38
7.1.8	RSI_SETPUBLICPAR()	38
7.1.9	RSI_RESET()	39
7.1.10	RSI_CHECKID()	39
7.1.11	RSI_ENABLE()	40
7.1.12	RSI_DISABLE()	40
7.2	Comportement des instructions RSI	40
7.2.1	Comportement RSI_ENABLE()/RSI_DISABLE()	40
7.2.2	Comportement RSI_ON()/RSI_OFF()	41
7.3	Programmation du traitement des signaux	42
7.3.1	Liaison du flux de signaux dans le programme KRL	42
7.3.2	Modification des paramètres de flux de signaux dans KRL	43
7.4	Configuration du fichier XML pour la liaison Ethernet	43
7.4.1	Structure XML des propriétés de liaison	44
7.4.2	Structure XML pour l'envoi de données	44
7.4.3	Structure XML de la réception de données	46
7.4.4	Configuration selon le schéma XML	48
7.4.5	Mots clés - Lecture des données	48
7.4.6	Mots clés - Ecriture des données	49
8	Exemples	51
8.1	Configurations et de programmes à titre d'exemple	51
8.1.1	Implémentation de l'application à titre d'exemple	52
8.1.2	Interface utilisateur du programme de serveur	52
8.1.3	Réglage des paramètres de communication dans le programme de serveur	54
8.1.4	Exemple de correction cartésienne via Ethernet	55
8.1.5	Exemple de mouvement circulaire guidé par un capteur	56
8.1.6	Exemple de correction de trajectoire par rapport à la commande de distance	59
8.1.7	Exemple d'une transformation dans un nouveau système de coordonnées	62
9	Diagnostic	67
9.1	Affichage de données de diagnostic concernant le RSI	67
9.2	Protocole de défauts (table de messages)	67
9.2.1	Configuration du LOG-Level	68
10	Messages	69
10.1	Messages en cours de fonctionnement	69
11	Annexe	71
11.1	Augmentation de la mémoire	71
11.2	Bibliothèque d'objets RSI	71
11.2.1	Objets RSI pour surveillance de la correction	71
11.2.2	Objets RSI pour transmission des signaux	71
11.2.3	Objets RSI pour transformation des coordonnées	72
11.2.4	Objets RSI pour liaison logique	72

11.2.5	Objets RSI pour liaison logique binaire	72
11.2.6	Objets RSI pour comparaisons mathématiques	73
11.2.7	Objets RSI pour opérations mathématiques	73
11.2.8	Objets RSI pour commande des signaux	73
11.2.9	Autres objets RSI	74
11.2.10	Objets RSI pour actions	75
12	SAV KUKA	77
12.1	Demande d'assistance	77
12.2	Assistance client KUKA	77
	Index	85

1 Introduction

1.1 Groupe cible

Cette documentation s'adresse aux utilisateurs disposant des connaissances suivantes :

- Connaissances approfondies en matière de programmation KRL
- Connaissances approfondies du système du contrôleur de robot
- Connaissances approfondies des systèmes de bus
- Connaissances de base en XML
- Connaissances de base en technique numérique



Pour une application optimale de nos produits, nous recommandons à nos clients une formation au KUKA College. Consultez notre site Internet www.kuka.com ou adressez-vous à une de nos filiales pour tout complément d'information sur notre programme de formation.

1.2 Documentation du robot industriel

La documentation du robot industriel est formée des parties suivantes :

- Documentation pour l'ensemble mécanique du robot
- Documentation pour la commande de robot
- Manuels de service et de programmation pour le logiciel système
- Instructions relatives aux options et accessoires
- Catalogue des pièces sur support de données

Chaque manuel est un document individuel.

1.3 Représentation des remarques

Sécurité

Ces remarques se réfèrent à la sécurité et **doivent** donc être respectées impérativement.



DANGER

Ces remarques signifient que des blessures graves, voire même mortelles vont sûrement ou très vraisemblablement **être** la conséquence de l'absence de mesures de précaution.



AVERTISSEMENT

Ces remarques signifient que des blessures graves, voire même mortelles **peuvent être** la conséquence de l'absence de mesures de précaution.



ATTENTION

Ces remarques signifient que des blessures légères **peuvent être** la conséquence de l'absence de mesures de précaution.

AVIS

Ces remarques signifient que des dommages matériels **peuvent être** la conséquence de l'absence de mesures de précaution.



Ces remarques renvoient à des informations importantes pour la sécurité ou à des mesures de sécurité générales.
Ces remarques ne se réfèrent pas à des dangers isolés ou à des mesures de sécurité individuelles.

Cette remarque attire l'attention sur des procédures permettant d'éviter ou d'éliminer des cas d'urgence ou de panne :

**INSTRUCTIONS
DE SÉCURITÉ**

Les procédures caractérisées par cette remarque **doivent** être respectées avec précision.

Remarques

Ces remarques facilitent le travail ou renvoient à des informations supplémentaires.



Remarque facilitant le travail ou renvoi à des informations supplémentaires.

1.4 Termes utilisés

Termes RSI

Terme	Description
RSI	Robot Sensor Interface Interface de communication entre le robot industriel et le système de capteurs
Conteneur RSI	Un conteneur RSI renferme le flux de signaux configuré avec RSI Visual et doit être créé dans le programme KRL.
ID de conteneur RSI	Identificateur attribué automatiquement dans le programme KRL lors de la création du conteneur RSI.
Contexte RSI	Le contexte RSI est le flux de signaux configuré avec RSI Visual, il est composé d'objets RSI et de liens entre les objets RSI.
RSI Monitor	Ecran de visualisation en ligne de signaux RSI
Objet RSI	Le flux de signaux est configuré au moyen d'objets RSI liés par le biais d'entrées et de sorties spécifiques à l'objet.
Bibliothèque d'objets RSI	Bibliothèque contenant tous les objets disponibles pour configurer le flux de signaux dans RSI Visual.
Paramètres d'objet RSI	Les paramètres d'objet RSI déterminent la fonctionnalité d'un objet RSI. Le nombre de paramètres d'objet RSI est spécifique à chaque objet RSI.
RSI Visual	Editeur graphique permettant de configurer le flux de signaux (contexte RSI)

Termes généraux

Terme	Description
CCS	Correction Coordinate System Système de coordonnées de correction au CDO pour la correction de capteur cartésienne
Ethernet	Ethernet est une technologie de réseau de données pour réseaux de données locaux (LAN). Elle permet d'échanger des données sous forme de cadres de données entre les participants raccordés.

Terme	Description
KLI	KUKA Line Interface Bus de ligne permettant d'intégrer l'installation dans le réseau du client
KR C	KUKA Roboter Controller
KUKA smartHMI	KUKA smart Human-Machine Interface Interface utilisateur du logiciel KUKA System Software
Mode Capteur	Mode de traitement des signaux <ul style="list-style-type: none"> ■ IPO : Traitement des signaux en cycle de capteur de 12 ms ■ IPO_FAST : Traitement des signaux en cycle de capteur de 4 ms
Cycle de capteur	Cycle utilisé pour calculer le traitement du signal. Selon le mode, le cycle du capteur est de 12 ms (mode IPO) ou de 4 ms (mode IPO_FAST)
TTS	Toolbased Technological System Le TTS est un système de coordonnées qui suit la trajectoire. Il est calculé à chaque déplacement LIN ou CIRC. Il est formé par la tangente de trajectoire, le sens X+ du système de coordonnées TOOL et le vecteur perpendiculaire qui en résulte. Le système de coordonnées accompagnant la trajectoire est défini de la manière suivante : X_{TTS} : Tangente de la trajectoire Y_{TTS} : Vecteur perpendiculaire par rapport au plan à partir de la tangente de la trajectoire dans le sens X du système de coordonnées TOOL. Z_{TTS} : Vecteur du système de droite formé à partir de X_{TTS} et Y_{TTS} La tangente de trajectoire et le sens +X du système de coordonnées TOOL ne doivent pas être parallèles, le TTS ne pouvant sinon pas être calculé.
UDP	User Datagram Protocol Protocole sans connexion relatif à la transmission des données entre les usagers d'un réseau.
IP	Internet-Protocol Le protocole Internet a pour tâche de définir des sous-réseau via des adresses MAC physiques.
XML	Extensible Markup Language Standard de création de documents lisibles par les machines et les opérateurs sous forme d'une structure arborescente prédéfinie

1.5 Marques

.NET Framework est une marque déposée de la Microsoft Corporation.

Visual Studio est une marque déposée par Microsoft Corporation.

Windows est une marque déposée par Microsoft Corporation.

2 Description du produit

2.1 Aperçu de RobotSensorInterface

- Fonctions** RobotSensorInterface est un progiciel technologique rechargeable doté des fonctions suivantes :
- Transmission de données entre le contrôleur du robot et le système de capteurs
 - Transmission de données via Ethernet ou le système E/S du contrôleur du robot
 - Traitement cyclique des signaux et analyse des signaux dans le cycle du capteur
 - Influence du mouvement du robot ou du déroulement du programme due au traitement des signaux de capteurs
 - Configuration du flux de signaux (contexte RSI) avec l'éditeur graphique RSI Visual
 - Bibliothèque d'objets RSI permettant de configurer le flux de signaux (contexte RSI)
 - Visualisation en ligne des signaux RSI (RSI Monitor)

Communication Le contrôleur du robot peut communiquer avec le système de capteurs via le système E/S ou via Ethernet.

Transmission des données via un système E/S :

- Les données et les signaux du système de capteurs sont lus et écrits via le système E/S. RobotSensorInterface accède aux données et aux signaux et les traite.



Les signaux sont liés avec le système E/S du contrôleur du robot par le biais d'un système de bus.

- Des informations générales sur la gestion de bus et la connexion E/S figurent dans la documentation de WorkVisual.
- Des informations détaillées sur la configuration de bus sont fournies dans la documentation du système de bus.

Transmission des données via Ethernet :

- Le contrôleur du robot communique avec le système de capteurs par le biais d'une liaison réseau travaillant en temps réel. Les données sont transmises via le protocole UDP/IP. Un cadre de données fixe n'est pas prescrit. L'utilisateur doit configurer le bloc de données dans un fichier XML.

Propriétés :

- Transmission de données cyclique du contrôleur du robot à un système de capteurs parallèlement au déroulement du programme (p. ex. données de position, angle d'axe, mode de fonctionnement, etc.)
- Transmission de données cyclique d'un système de capteurs au contrôleur du robot parallèlement au déroulement du programme (p. ex. valeurs de mesure de capteurs)



UDP est un protocole réseau sans connexion pour la transmission de paquets de données. La transmission de données via UDP n'est ni fiable, ni sûre. Par exemple, il n'est pas possible de garantir que des paquets transmis arrivent, ou qu'ils arrivent dans l'ordre dans lequel ils ont été envoyés.

La transmission des données via Ethernet est appropriée pour les applications insensibles au fait que des paquets soient perdus ou reçus dans le mauvais ordre. Si une application ne le tolère pas, le programmeur doit prendre des mesures appropriées, p. ex. contrôler dans le programme si tous les paquets sont arrivés, et redemander l'envoi des paquets si nécessaire.

2.2 Fonctionnement du traitement des signaux

Description

Un traitement de signal est établi avec des objets RSI. Avec ses entrées de signaux, un signal RSI exécute une fonctionnalité précise et délivre le résultat aux sorties de signaux.

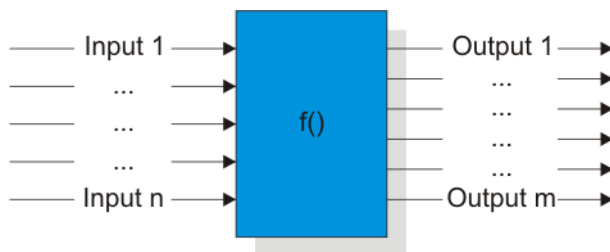


Fig. 2-1: Structure schématique d'un objet RSI

Dans une bibliothèque, RobotSensorInterface fournit à l'utilisateur un vaste jeu d'objets RSI. Un flux de signaux est créé en liant les entrées et sorties de signaux de plusieurs objets RSI. Le flux de signaux total est appelé « contexte RSI ».

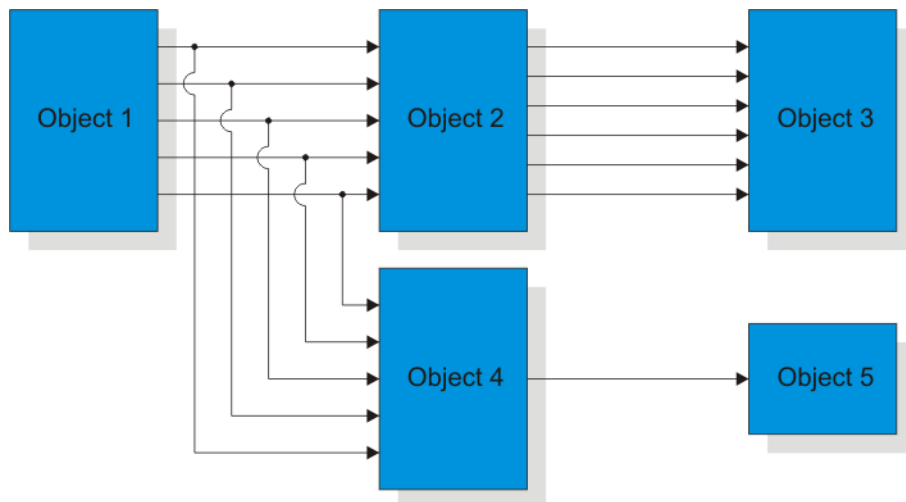


Fig. 2-2: Structure schématique d'un contexte RSI

Le contexte RSI est défini et mémorisé au moyen de l'éditeur graphique RSI Visual. Le contexte RSI peut être chargé et le traitement des signaux parallèle au déroulement du programme peut être activé et désactivé dans le programme KRL. Le traitement du signal est calculé dans le cycle du capteur. Selon le mode, le cycle du capteur est de 12 ms (mode IPO) ou de 4 ms (mode IPO_FAST).

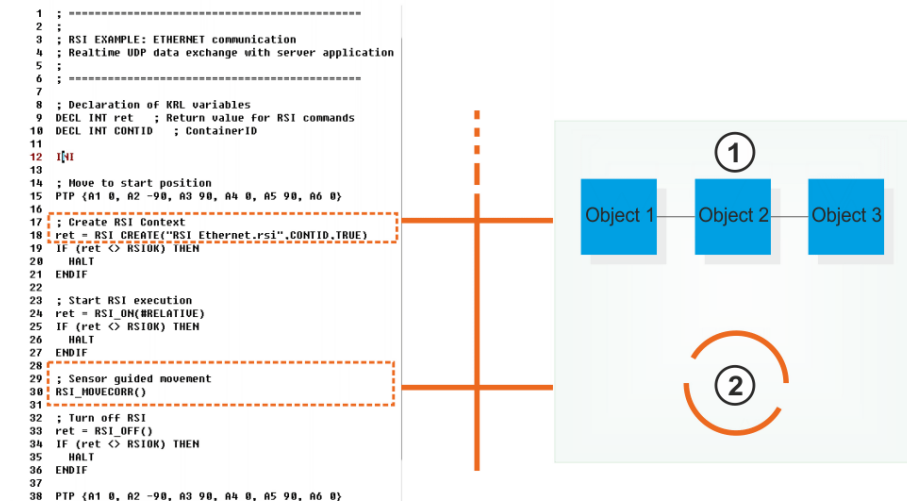


Fig. 2-3: Combinaison entre le programme KRL et le traitement des signaux

1 Contexte RSI

2 Cycle du capteur

2.3 Fonctionnement de la transmission des données

2.3.1 Transmission des données via un système E/S

Description

Les données et les signaux du système de capteurs sont lus via le système E/S (\$IN, \$ANIN) du contrôleur du robot. Les signaux traités sont retournés au système de capteurs via le système E/S (\$OUT, \$ANOUT). Les signaux sont lus et écrits dans le cycle du capteur.

Les objets RSI suivants sont utilisés :

- ANIN et DIGIN interviennent en lecture sur le système E/S et transmettent les données et les signaux du système de capteurs au traitement des signaux.
- MAP2ANOUT et MAP2DIGOUT accèdent aux signaux traités et les écrivent sur le système E/S.

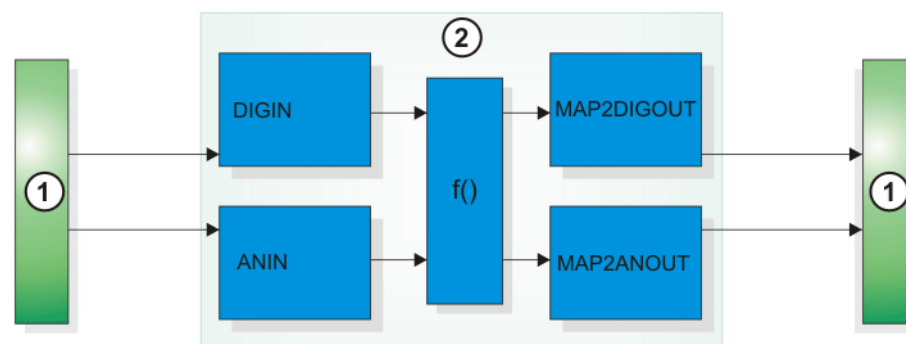


Fig. 2-4: Transmission des données via un système E/S

1 Système E/S

2 Contexte RSI

2.3.2 Transmission des données via Ethernet

Description

La transmission des données via Ethernet est réalisée au moyen de l'objet RSI ETHERNET.

Il est possible de définir 64 entrées et sorties pour ETHERNET. Les signaux sur les entrées sont transmis au système de capteurs. Les données reçues par le système de capteurs sont disponibles sur les sorties.

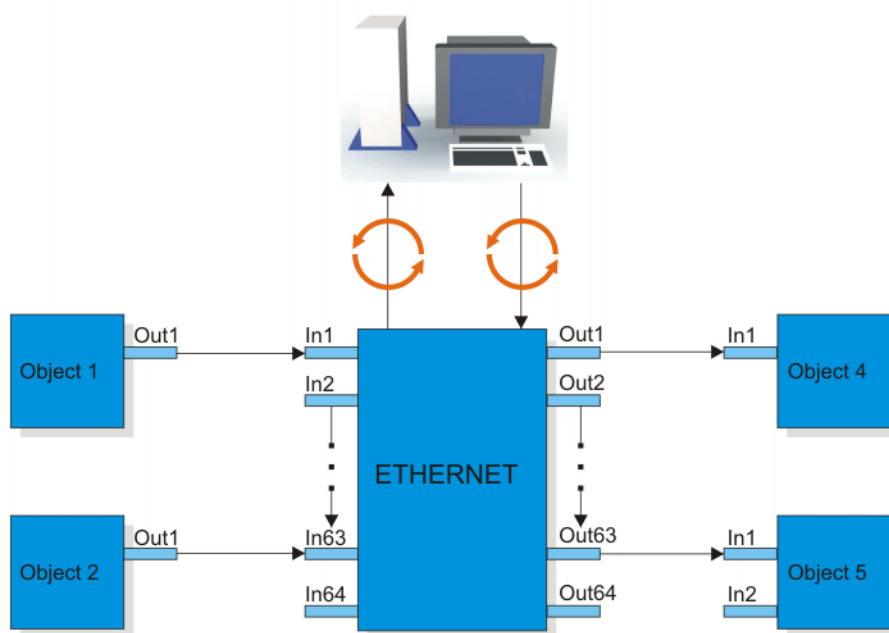


Fig. 2-5: Transmission des données via Ethernet (principe de fonctionnement)

Lorsque le traitement des signaux est activé, un canal qui transmet des données au système de capteurs via le protocole UDP/IP est préparé. Le contrôleur du robot initialise la transmission des données par un paquet de données, et transmet d'autres paquets de données au système de capteurs dans le cycle du capteur. Le système de capteurs doit répondre au paquet de données reçues en transmettant son propre paquet de données.

Lorsque le traitement des signaux est activé, ETHERNET transmet et reçoit dans le cycle du capteur un bloc de données au format XML défini par l'utilisateur. Ce bloc de données doit être configuré dans un fichier XML. Le nom du fichier XML est indiqué dans l'objet ETHERNET.

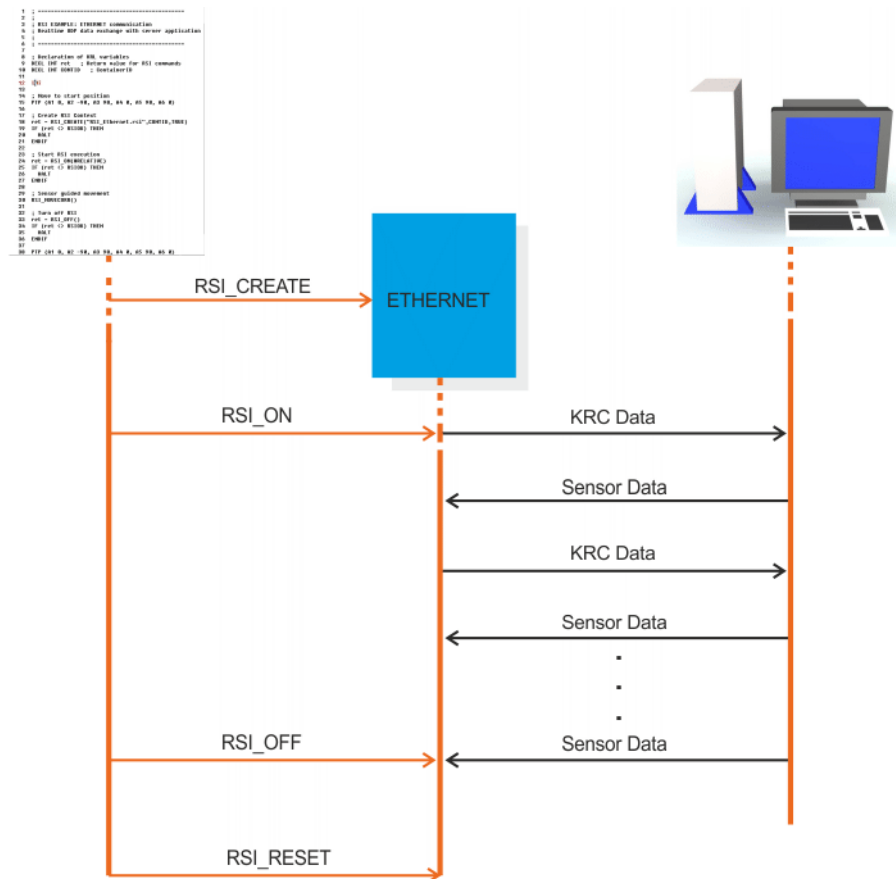


Fig. 2-6: Transmission des données via Ethernet (déroulement)

Demande en temps réel

Un paquet de données entrant sur le système de capteurs doit recevoir une réponse dans les délais du cycle du capteur. Les paquets qui arrivent trop tard sont rejetés.

Le robot s'arrête si le nombre maximum de paquets des données auxquels il a été répondu trop tard est dépassé. La transmission des données s'arrête lorsque le traitement des signaux est désactivé.

2.4 Mode de fonctionnement de la correction de capteur

La correction de capteur ne peut pas être employée sur les axes synchrones.

Aperçu

RobotSensorInterface permet d'exercer une influence continue sur les mouvements du robot via les données des capteurs. Lors de cette opération, une valeur de correction est calculée sur la position de consigne courante dans le cycle du capteur.

Les types de corrections suivants peuvent être configurés :

- Mouvements avec correction de capteur superposée :
 - Correction d'angle d'axe absolue ou relative
 - Correction cartésienne absolue ou relative
- Mouvement guidé par capteur :
 - Correction d'angle d'axe absolue ou relative
 - Correction cartésienne absolue ou relative

AVIS

Les corrections de capteurs influencent directement le mouvement du robot. C'est le capteur qui définit la trajectoire et non pas le robot industriel. L'utilisateur est responsable du traitement des demandes de correction du capteur sur le plan de la technique des signaux, de sorte que le système de robot ne puisse subir aucun dommage mécanique, dû p. ex. à des oscillations.

Correction de l'angle d'axe

Une valeur de correction peut être activée axe par axe sur les axes de robot A1 à A6 et les axes supplémentaires E1 à E6.

Objets RSI utilisés :

- AXISCORR (correction des axes du robot)
- AXISCORREXT (correction des axes supplémentaires)

La correction maximale admissible dans les deux sens est limitée.

Correction cartésienne

Une valeur de correction (Frame) permet de déplacer la position du robot de manière cartésienne. Le cadre de correction se rapporte à un système de coordonnées de correction (CCS) dans le CDO.

Les systèmes de coordonnées de référence suivants sont disponibles pour orienter le système de coordonnées de correction :

- Système de coordonnées BASE
- Système de coordonnées ROBROOT
- Système de coordonnées TOOL
- Système de coordonnées WORLD
- Tripode technologique (TTS)

Objet RSI utilisé :

- POSCORR

La correction cartésienne maximale autorisée est limitée.



Si RobotSensorInterface est utilisé dans RoboTeam, tenir compte du fait que le robot maître ne peut pas transmettre des corrections de capteurs cartésiennes au robot esclave.

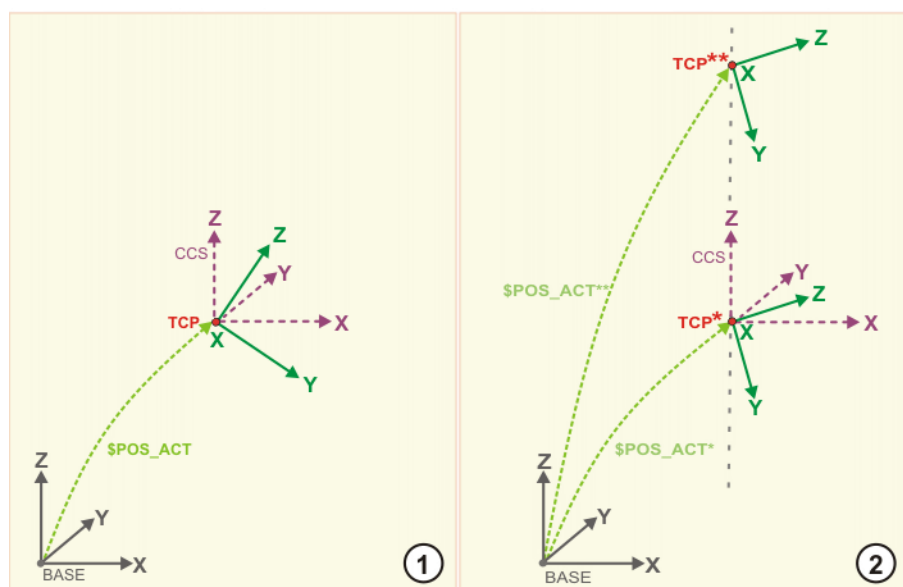


Fig. 2-7: Correction cartésienne se rapportant à BASE

Pos.	Description
1	Position de départ de la correction cartésienne <ul style="list-style-type: none"> ■ $\\$POS_ACT$: position cartésienne du robot ■ CCS : système de coordonnées de correction dans le CDO avec l'orientation BASE
2	Correction cartésienne - le système de coordonnées de correction est le système de coordonnées BASE <ul style="list-style-type: none"> ■ $\\$POS_ACT^*$: position cartésienne du robot pivotée de la valeur de correction <ul style="list-style-type: none"> ■ TCP* : le CDO est pivoté de +B dans le système de coordonnées de correction ■ $\\$POS_ACT^{**}$: position cartésienne du robot décalée et pivotée de la valeur de correction <ul style="list-style-type: none"> ■ TCP** : le CDO est décalé dans le sens +Z et pivoté de +B dans le système de coordonnées de correction

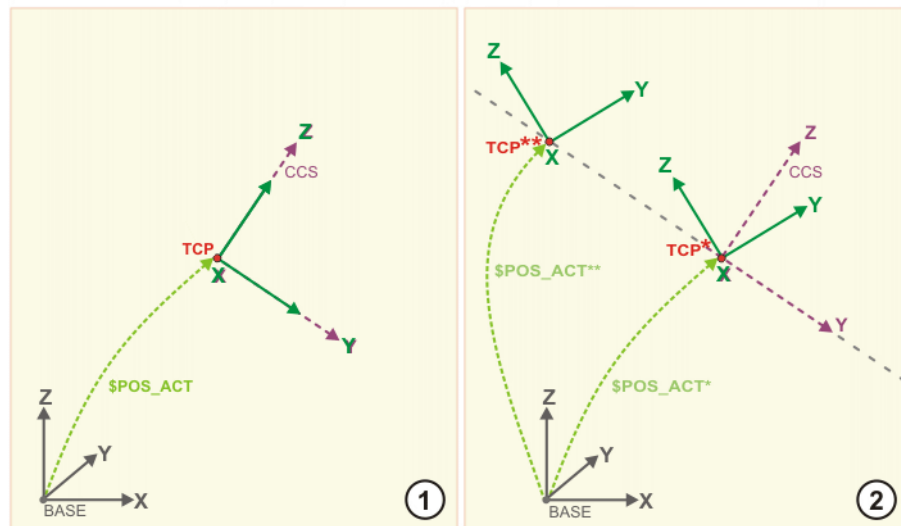


Fig. 2-8: Correction cartésienne se rapportant à TOOL

Pos.	Description
1	Position de départ de la correction cartésienne <ul style="list-style-type: none"> ■ $\\$POS_ACT$: position cartésienne du robot ■ CCS : système de coordonnées de correction dans le CDO avec l'orientation de TOOL
2	Correction cartésienne - le système de coordonnées de correction est le système de coordonnées TOOL <ul style="list-style-type: none"> ■ $\\$POS_ACT^*$: position cartésienne du robot pivotée de la valeur de correction <ul style="list-style-type: none"> ■ TCP* : le CDO est pivoté de +C dans le système de coordonnées de correction. ■ $\\$POS_ACT^{**}$: position cartésienne du robot décalée et pivotée de la valeur de correction <ul style="list-style-type: none"> ■ TCP** : le CDO est décalé dans le sens -Y et pivoté de +C dans le système de coordonnées de correction.

Correction absolue

La nouvelle position résulte du décalage de la position de départ correspondant à la valeur de correction courante.

Correction relative

Les valeurs de correction sont additionnées. La nouvelle position résulte du décalage de la position de départ correspondant à la correction antérieure et à la valeur de correction courante.

Correction de capteur superposée

Les valeurs de correction sont activées sur les points d'appui d'une trajectoire programmée. La trajectoire peut être corrigée sur la base de valeurs de correction absolues ou relatives.



Lorsque les signaux sont traités en mode IPO, la trajectoire peut être corrigée uniquement sur la base des déplacements LIN et CIRC.

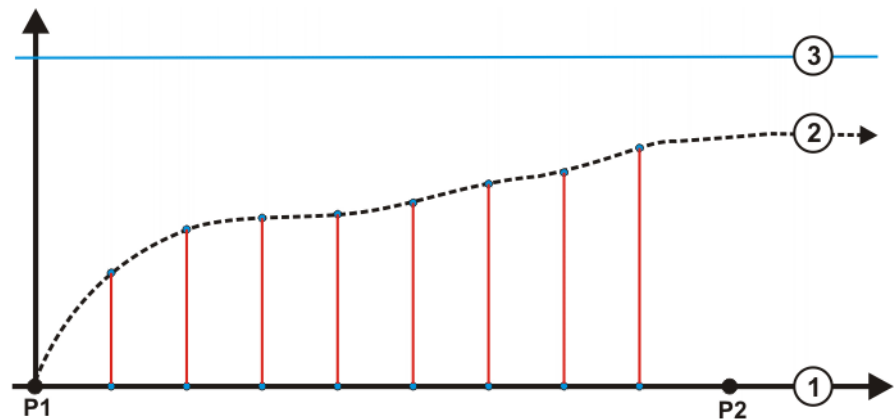


Fig. 2-9: Correction de trajectoire sur la base de valeurs absolues

- 1 Trajectoire programmée
- 2 Trajectoire corrigée
- 3 Correction totale maximale
- Rouge Valeur de correction absolue

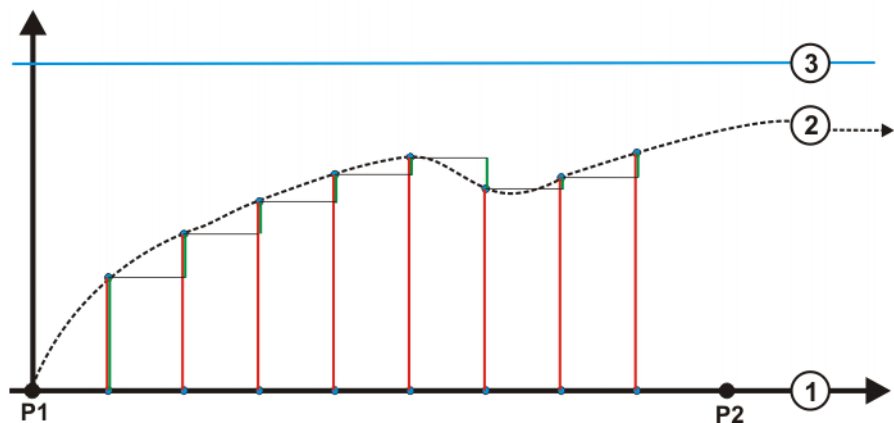


Fig. 2-10: Correction de trajectoire sur la base de valeurs relatives

- 1 Trajectoire programmée
- 2 Trajectoire corrigée
- 3 Correction totale maximale
- Rouge Correction totale
- Verte Valeur de correction relative

Mouvement guidé par capteur

L'instruction RSI_MOVECORR() permet de programmer un mouvement guidé par capteur. A partir d'un point de départ, le robot n'accoste pas un point de

destination défini, mais se déplace uniquement selon la commande de correction sur la base de données de capteur.

Le mouvement guidé par capteur peut être exécuté sur la base de valeurs de correction absolues ou relatives.

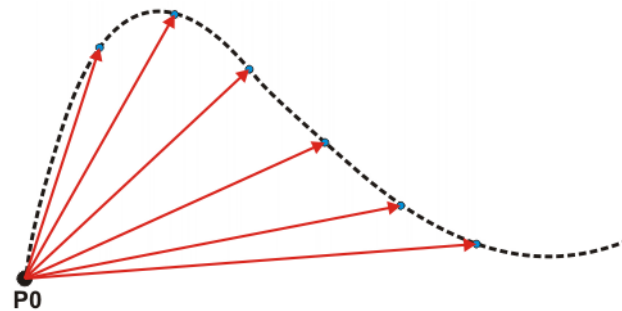


Fig. 2-11: Mouvement guidé par capteur sur la base de valeurs absolues

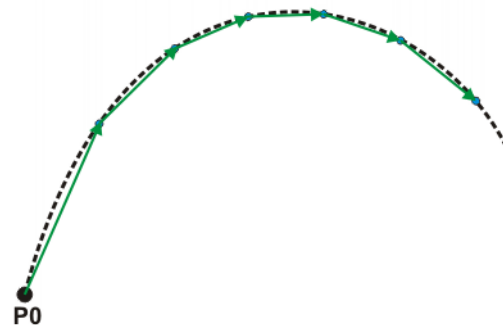


Fig. 2-12: Mouvement guidé par capteur sur la base de valeurs relatives



Le comportement du robot lors d'un déplacement guidé par capteur se différencie des KR C2 et KR C4. Pour une conversion des installations correspondantes de KR C2 vers KR C4, adressez-vous à l'assistance KUKA.

3 Sécurité

Cette documentation contient des remarques relatives à la sécurité se référant de façon spécifique au logiciel décrit ici.

Les informations fondamentales relatives à la sécurité concernant le robot industriel peuvent être consultées dans le chapitre "Sécurité" du manuel de service et de programmation pour intégrateurs de systèmes ou dans le manuel de service et de programmation pour l'utilisateur final.



Il est impératif de respecter le chapitre « Sécurité » du manuel de service et de programmation de KUKA System Software (KSS). Un danger de mort, un risque de blessures graves ou de dommages matériels importants pourraient sinon s'ensuivre.

3.1 Remarques relatives à la sécurité

Mode Capteur

- Des dommages corporels et matériels peuvent se produire si RobotSensorInterface est mal utilisé.
- En mode Capteur, le robot peut se déplacer de façon inattendue dans les cas suivants :
 - Objets RSI mal paramétrés
 - En cas de défaut matériel (p. ex. mauvais câblage, rupture du câble du capteur ou dysfonctionnement du capteur)
- Des mouvements inattendus peuvent provoquer de graves blessures et des dommages matériels importants. L'intégrateur de système est tenu de minimiser le risque de blessures pour lui et des tiers ainsi que le risque de dommages matériels en prenant des mesures de sécurité adéquates, p. ex. en limitant l'enveloppe d'évolution.
- Lorsque le traitement des signaux est lancé avec RobotSensorInterface, la commande de sécurité émet un message à acquitter en mode T1 ou T2 :

!!! Attention - La correction de capteur est activée !!!

Limitation de l'enveloppe d'évolution

- Les enveloppes de tous les axes du robot sont limitées par des butées logicielles réglables. Ces butées logicielles doivent être réglées de façon à ce que l'enveloppe d'évolution du robot soit limitée à la zone minimum nécessaire pour le processus.
- Le logiciel System Software permet de configurer un maximum de 8 enveloppes d'évolution cartésiennes et 8 enveloppes d'évolution spécifiques aux axes. L'intégrateur de système doit configurer les enveloppes d'évolution de façon à ce qu'elles soient limitées à la zone minimum nécessaire pour le processus. Ceci permet de réduire à un minimum le risque de dommages provoqués par des déplacements inattendus en mode capteur.



Pour tout complément d'informations sur la configuration d'enveloppes d'évolution, prière de consulter le manuel de service et de programmation pour intégrateurs de systèmes.

Correction de capteur

RobotSensorInterface surveille et limite la correction de capteur maximale. Chaque objet de correction peut être surveillé séparément, ainsi que la correction globale de tous les objets de correction.

Les corrections de capteurs se rapportant à l'objet sont limitées en standard à +/- 5 mm ou 5°, la correction globale à maximum +/- 6 mm ou 6°.

Si une correction se rapportant à l'objet est dépassée, le traitement des signaux se poursuit et la correction est limitée automatiquement à la correction

maximale autorisée. Le traitement du signal est arrêté si la correction globale autorisée est dépassée.

4 Installation

4.1 Conditions requises par le système

- Matériel**
- Contrôleur de robot KR C4
 - Pour la transmission de données via Ethernet :
 - Système externe assisté par microprocesseur doté d'un système d'exploitation et d'une carte réseau capables de travailler en temps réel à 100 M-Octets en mode Duplex intégral.
 - Capteur assisté par microprocesseur avec carte réseau capable de travailler en temps réel pour applications sensorielles.
 - Câble réseau pour Switch, couse ou câble réseau croisé pour la connexion directe
 - Pour la transmission de données via un système E/S : Système de bus, p. ex. ProfiNet
 - PC externe pour configurer le flux de signaux avec RSI Visual

Robots conseillés RobotSensorInterface ne doit être utilisé qu'en combinaison avec des robots KUKA à six axes. Ne planifier la mise en œuvre d'autres robots qu'après avoir consulté KUKA Roboter GmbH.

(>>> 12 "SAV KUKA" Page 77)

- Logiciel**
- Contrôleur de robot :
- KUKA System Software 8.3 ou 8.4
- PC externe :
- Système d'exploitation Windows avec .Net Framework 3.5, y compris le Service Pack 1

Ressources KRL Les ressources KRL suivantes doivent être libres pour réaliser des corrections RSI en mode IPO :

Ressource KRL	Numéro
Générateur de fonctions	1

- Compatibilité**
- RobotSensorInterface ne doit pas être installé avec les progiciels technologiques suivants sur le même contrôleur de robot :
 - KUKA.ConveyorTech
 - KUKA.ServoGun TC
 - KUKA.ServoGun FC
 - KUKA.EqualizingTech
 - Si RobotSensorInterface et KUKA.RoboTeam sont installés sur le même contrôleur de robot, tenir compte du fait que le robot maître ne peut pas transmettre des corrections de capteur cartésiennes au robot esclave.

4.2 Installation ou mise à jour de RobotSensorInterface



Il est conseillé d'archiver toutes les données correspondantes avant la mise à jour d'un logiciel.

- Condition préalable**
- Logiciel sur la clé KUKA.USBData
 - Aucun programme n'est sélectionné.
 - Mode T1 ou T2

- Groupe d'utilisateur "Expert"

AVIS

Seule la clé KUKA.USBData peut être utilisée. Si une autre clé USB est utilisée, des données peuvent être perdues ou modifiées.

Procédure

1. Enficher la clé USB.
2. Dans le menu principal, sélectionner **Mise en service > Installer logiciel supplémentaire**.
3. Choisir **Nouveau logiciel**. Si un logiciel se trouvant sur la clé USB n'est pas affiché, appuyer sur **Rafraîchir**.
4. Marquer **RSI** et appuyer sur **Installer**. Confirmer la question de sécurité avec **Oui**. Les fichiers sont copiés sur le disque dur.
5. Si vous souhaitez installer un logiciel supplémentaire à partir de cette clé, répétez l'opération 4.
6. Retirer la clé USB.
7. Un nouveau démarrage peut être nécessaire, en fonction du logiciel supplémentaire. Dans ce cas, une demande de redémarrage est affichée. Confirmer avec **OK** et redémarrer le contrôleur du robot. L'installation est poursuivie et terminée.

Fichier LOG

Un fichier de protocole LOG est créé sous C:\KRC\ROBOTER\LOG.

4.3 Désinstallation de RobotSensorInterface



Il est conseillé d'archiver toutes les données correspondantes avant la désinstallation d'un logiciel.

Condition préalable

- Groupe d'utilisateurs Expert

Procédure

1. Dans le menu principal, sélectionner **Mise en service > Installer logiciel supplémentaire**. Tous les programmes supplémentaires installés sont affichés.
2. Marquer **RSI** et appuyer sur **Désinstaller**. Confirmer la question de sécurité avec **Oui**. La désinstallation est préparée.
3. Redémarrer le contrôleur du robot. La désinstallation est poursuivie et terminée.

Fichier LOG

Un fichier de protocole LOG est créé sous C:\KRC\ROBOTER\LOG.

4.4 Installation de RSI Visual sur un PC externe

Préparation

- Copier le dossier **RSIVisual** sur le PC externe :
 - Depuis la clé KUKA.USBData
 - Ou depuis le contrôleur du robot dans le répertoire D:\KUKA_OPT\RSI, si le logiciel est préinstallé.

Condition préalable

- Droits d'administrateurs locaux

Procédure

1. Lancer le programme **setup.exe** dans le dossier **RSIVisual**.
2. L'assistant d'installation de RSI Visual est ouvert. Suivre les instructions de l'assistant d'installation.

3. RSI Visual est installé en standard dans le dossier C:\Programme\KUKA Roboter GmbH\RSIVisual.
Sélectionner un répertoire différent si cela est souhaité.
4. Une fois l'installation terminée, cliquer sur **Close** afin de fermer l'assistant d'installation.

4.5 Désinstallation de RSI Visual

**Condition
préalable**

- Droits d'administrateurs locaux

Procédure

1. Dans le menu de démarrage Windows, sous **Paramètres > Panneau de configuration > Logiciel**, effacer l'entrée **RSIVisual**.
2. Effacer le dossier **RSIVisual** dans le répertoire C:\Programmes\KUKA Roboter GmbH.

5 Configuration

5.1 Connexion réseau via la KLI du contrôleur du robot

Description Une connexion réseau via la KLI du contrôleur du robot doit être établie pour l'échange de données via Ethernet. RSI nécessite pour ce faire son propre réseau de capteurs Ethernet, indépendant des autres sous-réseaux KLI.

En fonction de la spécification, les interfaces Ethernet suivantes sont disponibles en option à l'interface client du contrôleur du robot :

- Interface X66 (1 emplacement)
- Interface X67.1-3 (3 emplacements)



Pour tout complément d'information concernant les interfaces Ethernet, veuillez consulter le manuel ou les instructions de montage du contrôleur de robot.

5.2 Configuration du réseau de capteurs Ethernet

Condition préalable

- Groupe d'utilisateurs "Expert"
- Connexion réseau via la KLI du contrôleur de robot

Procédure

1. Dans le menu principal, sélectionner **Mise en service > Service > Réduire HMI**.
2. Dans le menu de démarrage Windows, sous **Tous les programmes**, sélectionner **RSI-Network**.
La fenêtre **Network Setup** s'ouvre. Les connexions réseau déjà installées sont affichées dans l'arborescence sous **Other Installed Interfaces**.
3. Dans l'arborescence, sous **RSI Ethernet**, marquer l'entrée **New** et appuyer sur **Edit**.
4. Entrer l'adresse IP et confirmer avec **OK**.



La zone d'adresse IP 192.168.0.x est verrouillée pour la configuration de la connexion réseau. L'adresse IP entrée doit se trouver dans son propre sous-réseau, il ne faut pas que l'adresse soit dans la zone d'adresse d'un autre sous-réseau KLI.

5. Redémarrer le contrôleur du robot avec un démarrage à froid.

5.3 Modification des variables locales dans le RSI.DAT

Des variables globales sont définies dans le fichier KRC:\R1\TP\RSI\RSI.DAT. Seules les variables décrites ici peuvent être modifiées.

Condition préalable

- Groupe d'utilisateurs Expert

Description

```
DEFDAT RSI PUBLIC
...
RSI global Variables:
GLOBAL BOOL RSIERRMSG=TRUE
...
; Flag for writing context information
GLOBAL INT RSITECHIDX=1
; Tech Channel used for RSI corrections
ENDDAT
```

Variable	Description
RSIERRMSG	<p>TRUE = les erreurs qui se produisent lors de l'exécution d'instructions RSI sont affichées avec un message de validation sur le smartHMI.</p> <p>FALSE = pas de message d'acquittement. Les valeurs de retour des instructions RSI doivent être analysées dans le programme KRL pour traiter les erreurs.</p> <p>Par défaut : TRUE</p>
RSITECHIDX	<p>Générateur de fonctions pour corrections RSI en mode IPO</p> <p>Valeur par défaut : 1</p> <p>Le nombre maximum de générateurs de fonctions est défini dans les paramètres de la machine (\$TECH_MAX).</p>

6 Commande

6.1 Aperçu de l'interface utilisateur RSI Visual

L'interface utilisateur est disponible dans les langues suivantes, selon le choix effectué lors de l'installation :

- Allemand
- Anglais

Par défaut, tous les éléments ne sont pas visibles sur l'interface utilisateur ; ils peuvent être affichés ou supprimés selon les besoins.

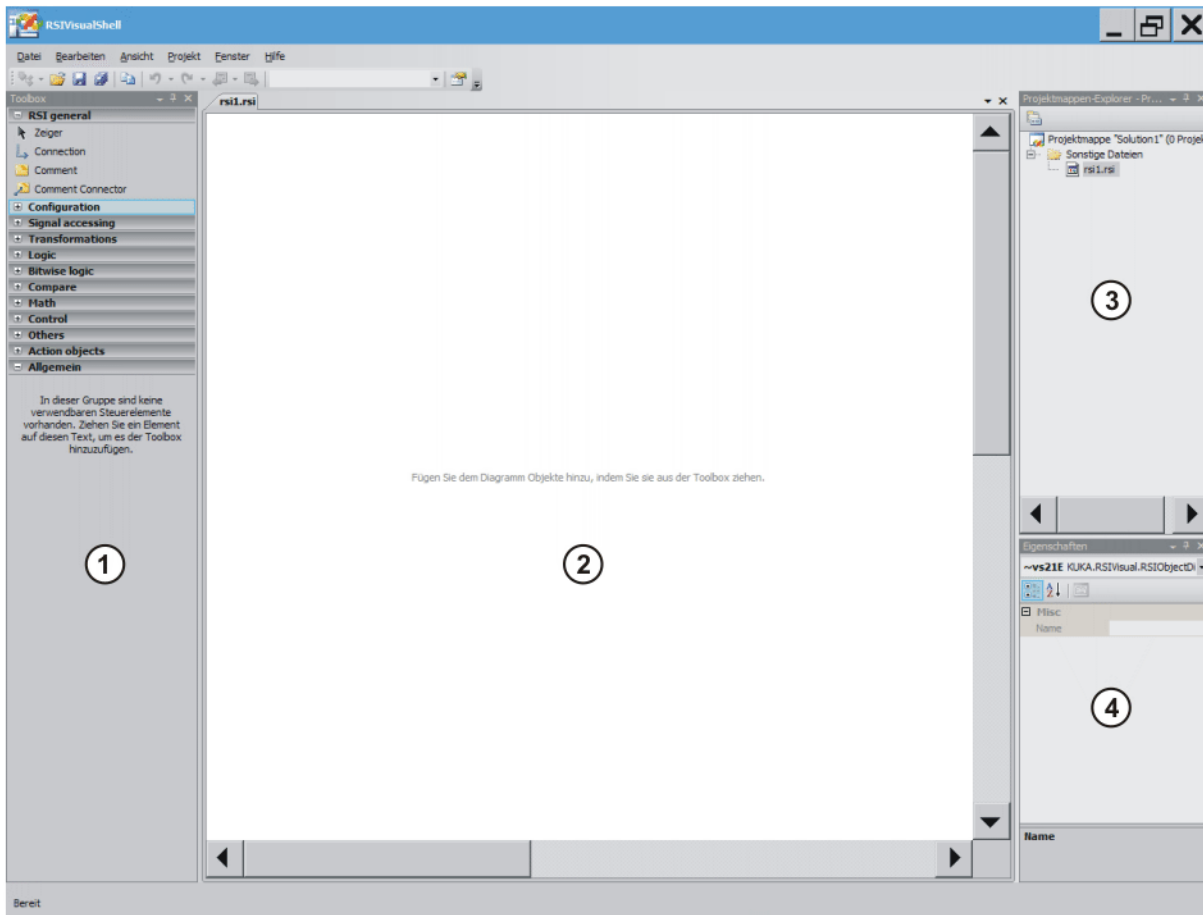


Fig. 6-1: Aperçu de l'interface utilisateur

Pos.	Description
1	<p>Fenêtre Toolbox</p> <p>Contient tous les outils et objets nécessaires pour configurer le contexte RSI. Les objets RSI peuvent être insérés dans l'éditeur de flux de signaux par Drag&Drop.</p> <p>Outils sous RSI general :</p> <ul style="list-style-type: none"> ■ Comment : un objet commentaire peut être inséré dans l'éditeur par Drag&Drop. ■ Comment Connector : un objet commentaire peut être lié avec l'objet RSI correspondant. <p>La description des objets RSI figure dans l'annexe. (>>> 11.2 "Bibliothèque d'objets RSI" Page 71)</p>
2	<p>Editeur de flux de signaux</p> <p>La configuration du flux de signaux est créée à cet endroit.</p>
3	<p>Fenêtre Explorateur de dossiers de projets</p> <p>Tous les fichiers chargés sont affichés dans une structure arborescente dans cette fenêtre.</p>
4	<p>Fenêtre Propriétés</p> <p>Lorsqu'un objet RSI, un paramètre d'objet RSI ou une sortie de signal d'entrée ou de sortie sont marqués dans l'éditeur de flux de signaux, leurs propriétés sont affichées. Il est possible de modifier des propriétés ou des paramètres individuels.</p>

6.1.1 Ouverture de l'éditeur de flux de signaux

Procédure

1. Sélectionner la séquence de menus **Fichier > Nouveau > Fichier....**
 2. Charger le modèle **rsi** avec **Ouvrir**.
- Un document vide est disponible pour configurer le flux de signaux.

6.1.2 Liaison des entrées de signaux d'entrée et de sortie



Description

Le flux de signaux est configuré au moyen des objets RSI qui sont insérés dans l'éditeur de flux de signaux par Drag&Drop et liés entre eux via les entrées et sorties de signaux spécifiques aux objets. Une sortie de signal peut être liée à plusieurs entrées de signaux.

Procédure

1. Pointer la sortie d'objet souhaitée avec le pointeur de la souris.
2. Lorsque le symbole de liaison est affiché sur la sortie, cliquer et indiquer la sortie d'objet souhaitée avec le pointeur de la souris.
3. Cliquer à nouveau lorsque le symbole de liaison est affiché sur l'entrée.

Symboles

Symbole	Description
	Symbole de liaison sur la sortie du signal
	Symbole de liaison sur l'entrée du signal



6.1.3 Insertion et liaison d'un commentaire

Procédure

1. Insérer un objet commentaire dans l'éditeur par Drag&Drop.
2. Marquer la zone de texte et saisir le commentaire.

3. Dans la **Toolbox**, sélectionner l'outil **Comment Connector**.
4. Pointer le commentaire avec le pointeur de la souris.
5. Lorsque le symbole de liaison est affiché sur le commentaire, cliquer et indiquer l'objet RSI souhaité avec le pointeur de la souris.
6. Cliquer à nouveau lorsque le symbole de liaison est affiché sur l'objet RSI.

Symboles

Symbole	Description
	Symbole de liaison sur le commentaire
	Symbole de liaison sur l'objet RSI

6.1.4 Réglage du paramètre d'objet RSI

Procédure

1. Marquer le paramètre d'objet RSI dans l'éditeur de flux de signaux.
Les propriétés du paramètre sont affichées dans la fenêtre **Propriétés**.
2. Entrer ou sélectionner la valeur souhaitée dans le champ **Value**.

6.1.5 Activation du paramètre d'objet RSI

Description

Il est possible de lire la valeur d'un paramètre d'objet RSI dans le programme KRL et d'assigner ultérieurement une nouvelle valeur au paramètre d'objet.

(>>> 7.3.2 "Modification des paramètres de flux de signaux dans KRL"
Page 43)

La condition pour ce faire est que le paramètre soit activé dans la configuration du flux de signaux.

Procédure

1. Marquer le paramètre d'objet RSI dans l'éditeur de flux de signaux.
Les propriétés du paramètre sont affichées dans la fenêtre **Propriétés**.
2. Commuter le champ **IsPublic** sur **True**.



Pour certains objets, l'activation de paramètres pendant que RSI est en cours n'est pas autorisée. Dans ce cas, le champ Feld IsPublic ne peut pas être commuté sur True.

6.1.6 Sauvegarde de la configuration du flux de signaux

Description

Les fichiers suivants sont créés lorsque la configuration du flux de signaux est sauvegardée :

- **<Nom du fichier>.rsi** : Configuration du flux de signaux de RSI Visual
- **<Nom du fichier>.rsi.diagram** : Structure du flux de signaux de RSI Visual selon le schéma XML
- **<Nom du fichier>.rsi.xml** : Fichier XML pour traitement des signaux sur le contrôleur du robot



Les fichiers RSI, DIAGRAM et XML constituent une unité et doivent être transmis ensemble au contrôleur du robot.

Répertoire cible :

C:\KRC\Roboter\Config\User\Common\SensorInterface

Procédure

1. Sélectionner la séquence de menu **Fichier > <Nom de fichier>Enregistrer .rsi** ou **<Nom de fichier>Enregistrer .rsi sous....**
2. Attribuer un nom à la configuration et la sauvegarder dans le répertoire souhaité avec **Enregistrer**.

6.1.7 Chargement de la configuration du flux de signaux

- Procédure**
1. Sélectionner la séquence de menus **Fichier > Ouvrir > Fichier....**
 2. Charger le fichier RSI souhaité avec **Ouvrir**.

6.2 Aperçu de l'interface utilisateur RSI Monitor

- Appel** ■ Dans le menu principal, sélectionner **Visualiser > RSI Monitor**.

Description RSI Monitor permet d'afficher et d'enregistrer jusqu'à 24 signaux à partir du contexte RSI. Pour ce faire, l'objet RSI MONITOR est utilisé dans le contexte RSI. Dans la configuration du flux de signaux, il faut que les signaux à afficher soient liés avec les entrées de l'objet MONITOR.

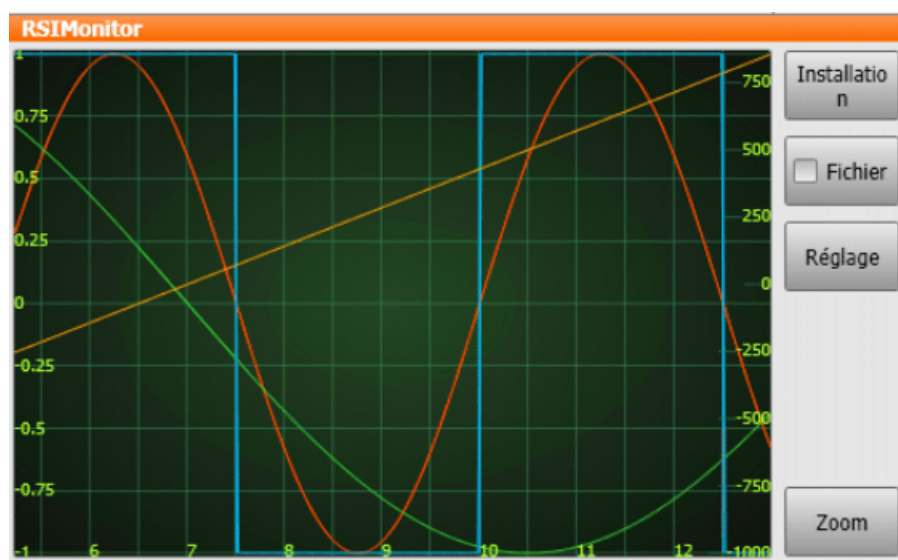


Fig. 6-2: Aperçu de l'interface utilisateur

Les boutons suivants sont disponibles :

Bouton	Description
Installation	Les propriétés des signaux pour l'enregistrement des signaux peuvent être définies.
Fichier	La séquence de signaux enregistrée peut être sauvegardée dans un fichier ou un fichier peut être chargé.
Réglage	Il est possible de régler le numéro de canal de l'objet RSI MONITOR. (Important si plusieurs objets MONITOR sont utilisés dans le contexte RSI.) ■ 1 ... 8 Par défaut : 1 Ce bouton n'est pas disponible pour le groupe « Utilisateur ».
Zoom	Le laps de temps peut être agrandi ou réduit à l'aide d'un régulateur. Le laps de temps visible peut être décalé en le tirant horizontalement sur la fenêtre d'affichage de l'écran.

6.2.1 Définition des propriétés des signaux

Description

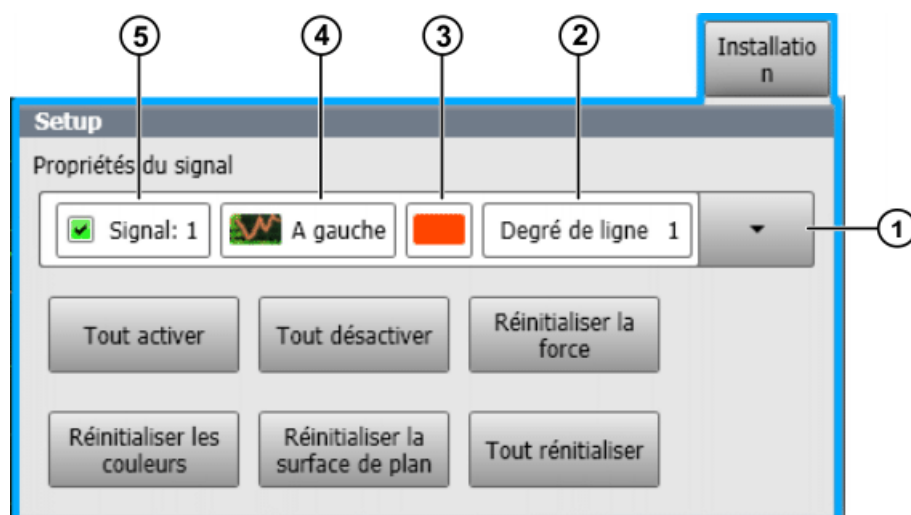


Fig. 6-3: RSI Monitor - Propriétés des signaux

Pos.	Description
1	Champ de liste avec les signaux 1 ... 24
2	Degré de ligne du signal ■ Degré de ligne1 ... 4 Par défaut : Degré de ligne1
3	Couleur du signal
4	Ordonnées auxquelles le signal se rapporte ■ A gauche : représentation se rapportant à l'ordonnée gauche. ■ A droite : représentation se rapportant à l'ordonnée droite. Par défaut : A gauche L'échelle des ordonnées est déterminée automatiquement en fonction du signal le plus puissant qui lui est assigné.
5	La case à cocher doit être activée (cochée) afin d'afficher un signal à l'écran. Par défaut : case à cocher activée pour Signal 1 ... 6 .

Les boutons suivants sont disponibles :

Bouton	Description
Tout activer	Active tous les signaux.
Tout désactiver	Désactive tous les signaux.
Réinitialiser la force	Ramène les degrés de lignes des signaux au degré de ligne 1 prédéfini.
Réinitialiser les couleurs	Ramène les couleurs des signaux aux couleurs prédéfinies.
Réinitialiser la surface de plan	Ramène la fenêtre temporelle à la taille prédéfinie (annuler le zoom)
Tout réinitialiser	Ramène tous les caractéristiques des signaux à la valeur prédéfinie.

6.2.2 Affichage de la forme des signaux

Description

Chaque objet MONITOR utilise son propre canal jusqu'au moniteur RSI. Si plusieurs objets MONITOR sont utilisés dans le contexte RSI, le numéro du

canal de l'objet MONITOR souhaité doit être réglé pour enregistrer le signal. Le moniteur RSI n'affiche que les signaux reçus via le canal réglé.

Condition préalable

- Adresse IP dans l'objet RSI MONITOR : 192.168.0.1

Procédure

1. Appeler le RSI Monitor et actionner **Installation**.
2. Régler les propriétés des signaux pour l'enregistrement.
3. Au besoin, passer dans le groupe d'utilisateurs Expert et appuyer sur **Réglage** pour régler le numéro de canal de l'objet MONITOR.
4. Sélectionner et traiter le programme.

L'enregistrement démarre lorsque le traitement des signaux est activé et se termine lorsqu'il est désactivé.



Un enregistrement de signal n'est supprimé dans le RSI Monitor que lorsqu'un objet MONITOR a été auparavant créé dans le programme KRL. L'enregistrement du signal est conservé dans RSI Monitor lorsque le programme est réinitialisé ou lorsque le traitement du signal est supprimé.

6.2.3 Sauvegarde de l'enregistrement des signaux

Procédure

1. Activer la case à cocher **Fichier**.
2. Affecter un nom de fichier à l'enregistrement dans le champ **Sauvegarder le fichier** et appuyer sur **Sauvegarder**.

L'enregistrement est sauvegardé en tant que fichier DAT dans le répertoire C:\KRC\ROBOTER\LOG\SensorInterface\MONITOR.

6.2.4 Chargement de l'enregistrement des signaux sur l'écran

Procédure

1. Activer la case à cocher **Fichier**.
2. Sélectionner le fichier souhaité dans le champ **Charger le fichier** et appuyer sur **Charger**.

Tous les enregistrements sauvegardés dans le répertoire C:\KRC\ROBOTER\LOG\SensorInterface\MONITOR sont disponibles au choix.

7 Programmation

7.1 Aperçu des instructions RSI

RobotSensorInterface propose des fonctions permettant de programmer le traitement des signaux. Chacune de ces fonctions, excepté RSI_MOVECORR(), possède une valeur de retour. La valeur de retour peut être interrogée et analysée dans le programme KRL.

Des constantes sont déclarées comme codes d'erreurs dans la liste de données RSI.DAT, dans le répertoire KRC:\R1\TP\RSI. Les constantes indiquées dans les descriptions des fonctions peuvent être utilisées pour vérifier si une instruction RSI a été exécutée correctement.

Fonctionnement	Description
RSI_CREATE()	(>>> 7.1.2 "RSI_CREATE()" Page 35)
RSI_DELETE()	(>>> 7.1.3 "RSI_DELETE()" Page 36)
RSI_ON()	(>>> 7.1.4 "RSI_ON()" Page 36)
RSI_OFF()	(>>> 7.1.5 "RSI_OFF()" Page 37)
RSI_MOVECORR()	(>>> 7.1.6 "RSI_MOVECORR()" Page 38)
RSI_GETPUBLICPAR()	(>>> 7.1.7 "RSI_GETPUBLICPAR()" Page 38)
RSI_SETPUBLICPAR()	(>>> 7.1.8 "RSI_SETPUBLICPAR()" Page 38)
RSI_RESET()	(>>> 7.1.9 "RSI_RESET()" Page 39)
RSI_CHECKID()	(>>> 7.1.10 "RSI_CHECKID()" Page 39)
RSI_ENABLE()	(>>> 7.1.11 "RSI_ENABLE()" Page 40)
RSI_DISABLE()	(>>> 7.1.12 "RSI_DISABLE()" Page 40)

7.1.1 Caractères et types

Les caractères et types suivants sont utilisés dans les descriptions de syntaxe :

Élément de syntaxe	Représentation
Code KRL	<ul style="list-style-type: none"> ■ Type Courier ■ Majuscules Exemples : GLOBAL; ANIN ON; OFFSET
Éléments devant être remplacés par des indications spécifiques au programme	<ul style="list-style-type: none"> ■ Italiques ■ Majuscules / minuscules Exemples : <i>Distance</i> ; <i>temps</i> ; <i>format</i>
Éléments en option	<ul style="list-style-type: none"> ■ Entre crochets pointus Exemple : <STEP <i>Divisions</i> >
Éléments s'excluant mutuellement	<ul style="list-style-type: none"> ■ Séparés par le caractère " " Exemple : IN OUT

7.1.2 RSI_CREATE()

Description

RSI_CREATE() crée un conteneur RSI et charge le flux de signaux configuré avec RSI Visual dans le conteneur. L'ID du conteneur permet d'accéder au conteneur créé.

Le conteneur créé avec RSI_CREATE() est activé par défaut. Si le conteneur est désactivé (élément *Etat* : IN), il faut le réactiver avec RSI_ON() avant d'activer le traitement du signal. RSI_ENABLE() active un conteneur désactivé.

Syntaxe

RET=RSI_CREATE(*Fichier* : *IN*<, *ID conteneur* : *OUT*><, *Etat* : *IN*>)

Explication de la syntaxe

Élément	Description
<i>RET</i>	Type : INT Valeurs retournées : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès ■ RSIFILENOTFOUND : fichier contenant la configuration du signal introuvable ■ RSIINVFILE : fichier invalide, p. ex. format de fichier invalide ou erreur dans la configuration ■ RSINOMEMORY : aucune mémoire RSI libre n'est plus disponible ■ RSIINVOBJTYPE : objet inconnu dans le contexte RSI ■ RSIEXTLIBNOTFOUND : bibliothèque d'objets externe introuvable ■ RSINOTLINKED : objet RSI avec signal d'entrée manquant ■ RSILNKCIRCLE : erreur dans le lien du flux de signaux
<i>Fichier</i> : <i>IN</i>	Type : champ CHAR Nom de la configuration du signal : <Nom du fichier>.rsi
<i>ID conteneur</i> : <i>OUT</i>	Type : INT ID du conteneur RSI
<i>Etat</i> : <i>IN</i>	Type : BOOL TRUE = active le conteneur RSI FALSE = désactive le conteneur RSI Par défaut : TRUE

7.1.3 RSI_DELETE()**Description**

RSI_DELETE() supprime un conteneur RSI et les objets RSI qu'il contient.

Syntaxe

RET=RSI_DELETE(*ID conteneur* : *IN*)

Explication de la syntaxe

Élément	Description
<i>RET</i>	Type : INT Valeurs retournées : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès ■ RSIINVOBJID : ID de conteneur invalide
<i>ID conteneur</i> : <i>IN</i>	Type : INT ID du conteneur RSI

7.1.4 RSI_ON()**Description**

Le traitement des signaux, le mode Correction et le mode Capteur sont activés avec RSI_ON().

Le traitement des signaux est exécuté par défaut dans le mode IPO_ FAST. Dans ce cas, il faut que le système de coordonnées de référence de la correc-

tion des capteurs soit configuré dans l'objet RSI POSCORR. Lorsque le traitement des signaux est activé en mode IPO, il faut que le système de coordonnées de référence soit défini avec RSI_ON().



Lorsque les signaux sont traités en mode IPO, la trajectoire peut être corrigée uniquement sur la base des déplacements LIN et CIRC.

Syntaxe

RET=RSI_ON(<mode Correction : IN><,mode Capteur : IN><,système de coordonnées : IN>)

Explication de la syntaxe

Elément	Description
<i>RET</i>	Type : INT Valeurs retournées : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès ■ RSIALREADYON : le traitement des signaux est déjà activé.
<i>Mode Correction : IN</i>	Type : ENUM Mode de correction : <ul style="list-style-type: none"> ■ #ABSOLUTE : Correction absolue ■ #RELATIVE : Correction relative Par défaut : #ABSOLUTE
<i>Mode Capteur : IN</i>	Type : ENUM Mode de traitement des signaux : <ul style="list-style-type: none"> ■ #IPO_FAST : 4 ms ■ #IPO : 12 ms avec filtrage (\$FILTER) Par défaut : #IPO_FAST
<i>Système de coordonnées : IN</i>	Type : ENUM Système de coordonnées de référence pour la correction de capteur (important uniquement si le mode Capteur = #IPO) <ul style="list-style-type: none"> ■ #BASE ■ #TCP ■ #TTS ■ #WORLD Par défaut : #BASE

7.1.5 RSI_OFF()

Description

RSI_OFF() désactive le traitement du signal.

Syntaxe

RET=RSI_OFF()

Explication de la syntaxe

Elément	Description
<i>RET</i>	Type : INT Valeurs retournées : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès ■ RSINOTRUNNING : pas de traitement de signal en cours

7.1.6 RSI_MOVECORR()

Description RSI_MOVECORR() active le mouvement guidé par capteur. Le robot est uniquement déplacé selon la commande de correction sur la base de données de capteurs, c.-à-d. avec les valeurs de correction des objets RSI POSCORR ou AXISCORR.

Un mouvement guidé par capteur peut être interrompu au moyen de l'objet RSI STOP.

Syntaxe RSI_MOVECORR(<Mode Stop : IN>)

Explication de la syntaxe

Élément	Description
<i>Mode Stop</i>	Type : ENUM Comportement à l'issue d'une interruption du mouvement : <ul style="list-style-type: none"> ■ #RSIBRAKE : le robot continue directement à partir du point d'arrêt. ■ #RSIBRAKERET : le robot revient au point de la trajectoire sur lequel le signal d'arrêt s'est produit. Par défaut : RSIBRAKE

7.1.7 RSI_GETPUBLICPAR()

Description La valeur de paramètre d'un objet RSI peut être lue avec RSI_GETPUBLICPAR(). Il faut pour ce faire que le paramètre de l'objet soit activé dans le contexte RSI.

Syntaxe RET=RSI_GETPUBLICPAR(ID conteneur : IN, Objet : IN, Paramètre : IN, Valeur : OUT)

Explication de la syntaxe

Élément	Description
<i>RET</i>	Type : INT Valeurs retournées : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès ■ RSIINVCONT : ID de conteneur invalide ■ RSIINPARAMID : nom d'objet ou de paramètre invalide, ou paramètre d'objet RSI non activé.
<i>ID conteneur : IN</i>	Type : INT ID du conteneur RSI
<i>Objet : IN</i>	Type : champ CHAR Nom de l'objet RSI
<i>Paramètre IN :</i>	Type : champ CHAR Nom du paramètre de l'objet RSI
<i>Valeur : OUT</i>	Type : REAL Valeur du paramètre de l'objet RSI

7.1.8 RSI_SETPUBLICPAR()

Description RSI_SETPUBLICPAR() permet d'assigner une nouvelle valeur au paramètre d'un objet RSI. Il faut pour ce faire que le paramètre de l'objet soit activé dans le contexte RSI.

Syntaxe

RET=RSI_SETPUBLICPAR(*ID conteneur : IN, Objet : IN, Paramètre : IN, Valeur : IN*)

Explication de la syntaxe

Élément	Description
<i>RET</i>	Type : INT Valeurs retournées : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès ■ RSIINVCONT : ID de conteneur invalide ■ RSIINPARAMID : nom d'objet ou de paramètre invalide, ou paramètre d'objet RSI non activé. ■ RSIINPARAM : valeur de paramètre d'objet RSI invalide
<i>ID conteneur : IN</i>	Type : INT ID du conteneur RSI
<i>Objet : IN</i>	Type : champ CHAR Nom de l'objet RSI
<i>Paramètre IN :</i>	Type : champ CHAR Nom du paramètre de l'objet RSI
<i>Valeur : IN</i>	Type : REAL Nouvelle valeur du paramètre de l'objet RSI

7.1.9 RSI_RESET()**Description**

RSI_RESET() supprime le traitement des signaux et tous les objets RSI.

Syntaxe

RET=RSI_RESET()

Explication de la syntaxe

Élément	Description
<i>RET</i>	Type : INT Valeur renvoyée : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès

7.1.10 RSI_CHECKID()**Description**

RSI_CHECKID() permet de vérifier si un ID de conteneur RSI valide est utilisé.

Syntaxe

RET=RSI_CHECKID(*ID conteneur : IN*)

Explication de la syntaxe

Élément	Description
<i>RET</i>	Type : BOOL Valeurs retournées : <ul style="list-style-type: none"> ■ TRUE = conteneur RSI correspondant à l'ID présent ■ TRUE = aucun conteneur RSI correspondant à l'ID présent
<i>ID conteneur : IN</i>	Type : INT ID du conteneur RSI

7.1.11 RSI_ENABLE()

Description RSI_ENABLE() active un conteneur RSI désactivé.

Syntaxe *RET*=RSI_ENABLE(*ID conteneur : IN*)

Explication de la syntaxe

Élément	Description
<i>RET</i>	Type : INT Valeurs retournées : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès ■ RSIINVOBJID : ID de conteneur invalide
<i>ID conteneur : IN</i>	Type : INT ID du conteneur RSI

7.1.12 RSI_DISABLE()

Description RSI_DISABLE() désactive un conteneur RSI.

Un conteneur désactivé doit être réactivé avant d'activer le traitement du signal avec RSI_ON(). RSI_ENABLE() active un conteneur désactivé.

Syntaxe *RET*=RSI_DISABLE(*ID conteneur : IN*)

Explication de la syntaxe

Élément	Description
<i>RET</i>	Type : INT Valeurs retournées : <ul style="list-style-type: none"> ■ RSIOK : fonction exécutée avec succès ■ RSIINVOBJID : ID de conteneur invalide
<i>ID conteneur : IN</i>	Type : INT ID du conteneur RSI

7.2 Comportement des instructions RSI

7.2.1 Comportement RSI_ENABLE()/RSI_DISABLE()

Description Les instructions RSI_ENABLE(ContId) et RSI_DISABLE(ContId) permettent de mettre en pause un contexte RSI déterminé. Les valeurs courantes au sein du contexte sont conservées à la suite du RSI_DISABLE. Le contexte RSI est poursuivi normalement à l'issue de la pause.



Fig. 7-1: Contexte RSI en pause

Exemple

Pause du contexte RSI avec RSI_DISABLE() lorsqu'un signal sinusoïdal est généré au moyen d'un objet SOURCE.

```
DEF RSI_DISABLEENABLE()
DECL INT ret,cont
ret=RSI_CREATE("Signals.rsi",cont())
ret=RSI_ON()
wait sec 7
ret=RSI_DISABLE(cont)
wait sec 1
ret=RSI_ENABLE(cont)
wait sec 10
ret=RSI_OFF()
END
```



Il est possible que les capteurs perdent leurs valeurs pendant la pause, ce qui peut provoquer un comportement inattendu lors du redémarrage du contexte avec un RSI_ENABLE(contId).

7.2.2 Comportement RSI_ON()/RSI_OFF()

Description

Les instructions RSI_OFF et RSI_ON permettent de redémarrer le contexte RSI complet. Ces instructions ne permettent pas d'appeler un contexte RSI individuel, mais uniquement l'ensemble des contextes générés par RSI_CREATE.



Fig. 7-2: Redémarrage du contexte

Exemple

Remise à zéro du contexte RSI avec RSI_OFF() lors de la génération d'un signal sinusoïdal par un objet SOURCE.

```
DEF RSI_ONOFF()
DECL INT ret,cont
ret=RSI_CREATE("Signals.rsi",cont)
ret=RSI_ON()
wait sec 7
ret=RSI_OFF()
wait sec 1
ret=RSI_ON()
Wait sec 10
ret=RSI_OFF()
END
```

7.3 Programmation du traitement des signaux

Aperçu

Etape	Description
1	Configurer le flux de signaux avec RSI Visual.
2	Transférer la configuration du flux de signaux (3 fichiers) sur le contrôleur du robot. Répertoire cible : C:\KRC\Roboter\Config\User\Common\SensorInterface
3	Lier le flux de signaux dans le programme KRL. (>>> 7.3.1 "Liaison du flux de signaux dans le programme KRL" Page 42)

7.3.1 Liaison du flux de signaux dans le programme KRL

Description

Le traitement des signaux doit être initialisé dans le programme KRL, activé, puis désactivé.

Structure d'un programme de traitement des signaux :

```
1 DEF signal_processing()
2
3 DECL INT ret
4
5 INI
6 ...
6 ret=RSI_CREATE("test.rsi")
7 ret=RSI_ON()
8 ...
10 movements
11 ...
15 ret=RSI_OFF()
16 ...
20 END
```

Ligne	Description
3	Déclaration des variables KRL (ici uniquement la variable « ret » pour la valeur de retour)
6	RSI_CREATE() initialise le traitement des signaux. La configuration du flux de signaux est chargée dans un conteneur RSI.
7	RSI_ON() active le traitement du signal.

Ligne	Description
10	Instructions de mouvement ou RSI_MOVECORR() pour un mouvement guidé par capteur
15	RSI_OFF() désactive le traitement du signal.

7.3.2 Modification des paramètres de flux de signaux dans KRL

Description Les paramètres de flux de signaux peuvent être modifiés à posteriori dans le programme KRL par le biais des fonctions suivantes :

- RSI_GETPUBLICPAR() : lit la valeur configurée d'un paramètre d'objet RSI
- RSI_SETPUBLICPAR() : assigne une nouvelle valeur au paramètre d'objet RSI

Condition préalable ■ Le paramètre d'objet RSI est activé.

Exemple (>>> 8.1.5 "Exemple de mouvement circulaire guidé par un capteur"
Page 56)

7.4 Configuration du fichier XML pour la liaison Ethernet

Aperçu RobotSensorInterface utilise le format XML pour échanger des données via Ethernet. Un fichier de configuration doit être défini dans le répertoire C:\KRC\ROBOTER\Config\User\Common\SensorInterface pour une liaison Ethernet.



RSI Visual contient le modèle **RSIEthernet** (séquence de menu **Fichier > Nouveau > Fichier...**). Le modèle peut être utilisé pour configurer la liaison Ethernet.

Le nom du fichier de configuration est indiqué dans l'objet ETHERNET de la configuration du flux de signaux et lu lors de l'initialisation du traitement des signaux dans le programme KRL.

```
<ROOT>
  <CONFIG></CONFIG>
  <SEND>
    <ELEMENTS></ELEMENTS>
  </SEND>
  <RECEIVE>
    <ELEMENTS></ELEMENTS>
  </RECEIVE>
</ROOT>
```

Paragraphe	Description
<CONFIG	Configuration des paramètres de liaison entre le système de capteurs et l'interface
...	
</CONFIG>	(>>> 7.4.1 "Structure XML des propriétés de liaison" Page 44)

Paragraphe	Description
<SEND> ... </SEND>	Configuration de la structure de transmission (>>> 7.4.2 "Structure XML pour l'envoi de données" Page 44)
<RECEIVE> ... </RECEIVE>	Configuration de la structure de réception (>>> 7.4.3 "Structure XML de la réception de données" Page 46)

7.4.1 Structure XML des propriétés de liaison

Description

Eléments de la structure XML :

Elément	Description
IP_NUMBER	Adresse IP du système de capteurs
PORT	Numéro de port du système de capteurs ■ 1 ... 65 534
SENTYPE	Identificateur du système de capteurs (nom au choix) Le contrôleur du robot vérifie l'identificateur à chaque paquet de données qu'il reçoit.
ONLYSEND	Sens de la transmission des données ■ TRUE = le contrôleur du robot transmet des données et n'attend aucune donnée du système de capteurs en retour. ■ FALSE = le contrôleur du robot transmet et reçoit des données. Par défaut : FALSE

Exemple

```
<CONFIG>
  <IP_NUMBER>172.1.10.5</IP_NUMBER>
  <PORT>49152</PORT>
  <SENTYPE>ImFree</SENTYPE>
  <ONLYSEND>FALSE</ONLYSEND>
</CONFIG>
```

7.4.2 Structure XML pour l'envoi de données

Description

A cet endroit sont définis les signaux issus du contexte RSI, qui arrivent aux entrées de l'objet ETHERNET et sont transmis au système de capteurs.

L'objet ETHERNET dispose en outre d'une fonction de lecture avec laquelle les informations système du contrôleur du robot peuvent être lues et transmises au système de capteurs. La fonction de lecture est activée par des mots clés.

RobotSensorInterface crée automatiquement le document XML transmis par le contrôleur du robot à partir de la structure XML configurée.

Entrées de signaux

Définition des entrées de signaux dans la structure XML :

Attribut	Description
TAG	Nom de l'élément La structure XML est ici définie pour la transmission des données (schéma XML). (>>> 7.4.4 "Configuration selon le schéma XML" Page 48)
TYPE	Type de données de l'élément <ul style="list-style-type: none"> ■ BOOL ■ DOUBLE ■ LONG
INDX	Numéro de l'entrée d'objet ETHERNET <ul style="list-style-type: none"> ■ 1 ... 64 <p>Remarque : Les entrées d'objets doivent être numérotées en continu.</p>

Exemple d'entrées de signaux

- Structure XML configurée pour la transmission de données :

```
<SEND>
<ELEMENTS>
  <ELEMENT TAG="Out.01" TYPE="BOOL" INDX="1" />
  <ELEMENT TAG="Out.02" TYPE="BOOL" INDX="2" />
  <ELEMENT TAG="Out.03" TYPE="BOOL" INDX="3" />
  <ELEMENT TAG="Out.04" TYPE="BOOL" INDX="4" />
  <ELEMENT TAG="Out.05" TYPE="BOOL" INDX="5" />
  <ELEMENT TAG="FTC.Fx" TYPE="DOUBLE" INDX="6" />
  <ELEMENT TAG="FTC.Fy" TYPE="DOUBLE" INDX="7" />
  <ELEMENT TAG="FTC.Fz" TYPE="DOUBLE" INDX="8" />
  <ELEMENT TAG="FTC.Mx" TYPE="DOUBLE" INDX="9" />
  <ELEMENT TAG="FTC.My" TYPE="DOUBLE" INDX="10" />
  <ELEMENT TAG="FTC.Mz" TYPE="DOUBLE" INDX="11" />
  <ELEMENT TAG="Override" TYPE="LONG" INDX="12" />
</ELEMENTS>
</SEND>
```

- Document XML transmis par le contrôleur du robot :

```
<Rob TYPE="KUKA">
  <Out 01="0" 02="1" 03="1" 04="0" 05="0" />
  <FTC Fx="1.234" Fy="54.75" Fz="345.7
    Mx="2346.6" My="12.0" Mz="3546" />
  <Override>90</Override>
  <IPOC>123645634563</IPOC>
</Rob>
```



Le mot clé IPOC transmet un horodatage et est créé automatiquement.

Fonction de lecture

Archivage de la fonction de lecture dans la structure XML :

Attribut	Description
TAG	Nom de l'élément Un mot clé indique l'information système qui sera lue. (>>> 7.4.5 "Mots clés - Lecture des données" Page 48)
TYPE	Type de données de l'élément <ul style="list-style-type: none"> ■ DOUBLE ■ LONG
INDX	Mot clé permettant de lire l'information système <ul style="list-style-type: none"> ■ INTERNAL

Exemple de fonction de lecture

(>>> "Exemple de fonction de lecture" Page 49)

7.4.3 Structure XML de la réception de données

Description

Là sont définis les signaux reçus par le système de capteurs sur les sorties de l'objet ETHERNET et transmis au contrôleur du robot dans le contexte RSI.

L'objet ETHERNET dispose en outre d'une fonction d'écriture qui permet d'écrire des informations dans le contrôleur du robot, ou de sortir des messages sur le SmartHMI. La fonction d'écriture est activée par un mot clé.

RobotSensorInterface crée automatiquement le document XML attendu par le contrôleur du robot à partir de la structure XML configurée.

Sorties de signaux

Définition des sorties de signaux dans la structure XML :

Attribut	Description
TAG	Nom de l'élément La structure XML est ici définie pour la réception des données (schéma XML). (>>> 7.4.4 "Configuration selon le schéma XML" Page 48)
TYPE	Type de données de l'élément <ul style="list-style-type: none"> ■ BOOL ■ DOUBLE ■ LONG
INDX	Numéro de la sortie d'objet ETHERNET <ul style="list-style-type: none"> ■ 1 ... 64 <p>Remarque : Les sorties d'objets doivent être numérotées en continu.</p>
HOLDON	Comportement de la sortie d'objet retenue dans le cas de l'arrivée tardive de paquets de données <ul style="list-style-type: none"> ■ 0 : la sortie est réinitialisée. ■ 1 : la dernière valeur valide entrée s'arrête à la sortie.

Exemple de sorties de signaux

- Structure XML configurée pour la réception de données :

```
<RECEIVE>
<ELEMENTS>
```

```

<ELEMENT TAG="RKorr.X" TYPE="DOUBLE" INDX="1" HOLDON="1" />
<ELEMENT TAG="RKorr.Y" TYPE="DOUBLE" INDX="2" HOLDON="1" />
<ELEMENT TAG="RKorr.Z" TYPE="DOUBLE" INDX="3" HOLDON="1" />
<ELEMENT TAG="RKorr.A" TYPE="DOUBLE" INDX="4" HOLDON="1" />
<ELEMENT TAG="RKorr.B" TYPE="DOUBLE" INDX="5" HOLDON="1" />
<ELEMENT TAG="RKorr.C" TYPE="DOUBLE" INDX="6" HOLDON="1" />
<ELEMENT TAG="AK.A1" TYPE="DOUBLE" INDX="7" HOLDON="0" />
<ELEMENT TAG="AK.A2" TYPE="DOUBLE" INDX="8" HOLDON="0" />
<ELEMENT TAG="AK.A3" TYPE="DOUBLE" INDX="9" HOLDON="0" />
<ELEMENT TAG="AK.A4" TYPE="DOUBLE" INDX="10" HOLDON="0" />
<ELEMENT TAG="AK.A5" TYPE="DOUBLE" INDX="11" HOLDON="0" />
<ELEMENT TAG="AK.A6" TYPE="DOUBLE" INDX="12" HOLDON="0" />
<ELEMENT TAG="EK.E1" TYPE="DOUBLE" INDX="13" HOLDON="0" />
<ELEMENT TAG="EK.E2" TYPE="DOUBLE" INDX="14" HOLDON="0" />
<ELEMENT TAG="EK.E3" TYPE="DOUBLE" INDX="15" HOLDON="0" />
<ELEMENT TAG="EK.E4" TYPE="DOUBLE" INDX="16" HOLDON="0" />
<ELEMENT TAG="EK.E5" TYPE="DOUBLE" INDX="17" HOLDON="0" />
<ELEMENT TAG="EK.E6" TYPE="DOUBLE" INDX="18" HOLDON="0" />
<ELEMENT TAG="DiO" TYPE="LONG" INDX="19" HOLDON="1" />
</ELEMENTS>
</RECEIVE>

```

■ Document XML reçu par le système de capteurs :

```

<Sen Type="ImFree">
  <RKorr X="4" Y="7" Z="32" A="6" B="" C="6" />
  <AK A1="2" A2="54" A3="35" A4="76" A5="567" A6="785" />
  <EK E1="67" E2="67" E3="678" E4="3" E5="3" E6="7" />
  <DiO>123</DiO>
  <IPOC>123645634563</IPOC>
</Sen>

```



L'horodatage accompagné du mot clé IPOC est contrôlé. Le paquet de données n'est valide que si l'horodatage correspond à l'horodatage transmis auparavant.

Fonction d'écriture

Archivage de la fonction d'écriture dans la structure XML :

Attribut	Description
TAG	Nom de l'élément Un mot clé indique l'information qui sera écrite dans le contrôleur du robot, ou si un message sera sorti sur le smartHMI.
TYPE	Type de données de l'élément <ul style="list-style-type: none"> ■ DOUBLE ■ STRING
INDX	Mot clé permettant d'écrire d'information <ul style="list-style-type: none"> ■ INTERNAL
HOLDON	Comportement de la sortie d'objet retenue dans le cas de l'arrivée tardive de paquets de données <ul style="list-style-type: none"> ■ 0 : la sortie est réinitialisée. ■ 1 : la dernière valeur valide entrée s'arrête à la sortie.

Exemple de fonction d'écriture

7.4.4 Configuration selon le schéma XML

Description

RobotSensorInterface crée automatiquement les documents XML pour la transmission des données à partir de la structure XML configurée.

On distingue les modes d'écriture suivants en fonction du schéma XML :

- Mode d'écriture d'élément
- Mode d'écriture d'attribut

Mode d'écriture d'élément

- TAG dans la structure XML configurée :

```
...
<ELEMENTS>
  <ELEMENT TAG="Out1" ... />
  <ELEMENT TAG="Out2" ... />
  <ELEMENT TAG="Out3" ... />
</ELEMENTS>
...
```

- TAG dans le document XML créé :

```
...
<Out1>...</Out1>
<Out2>...</Out2>
<Out3>...</Out3>
...
```

Mode d'écriture d'attribut

- TAG dans la structure XML configurée :

```
...
<ELEMENTS>
  <ELEMENT TAG="Out.01" ... />
  <ELEMENT TAG="Out.02" ... />
  <ELEMENT TAG="Out.03" ... />
</ELEMENTS>
...
```

- TAG avec attributs dans le document XML créé :

```
...
<Out 01="..." 02="..." 03="..." />
...
```

7.4.5 Mots clés - Lecture des données

Les mots clés sont des suites de lettres avec une signification constante attribuée. Ils ne peuvent pas être utilisés dans la structure XML au-delà de cette signification. L'écriture majuscule ou minuscule n'a pas d'importance. Un mot clé est valable en tant que mot clé quelle que soit l'écriture.

Mots clés

Les informations suivantes du contrôleur du robot peuvent être lues dans l'attribut TAG via des mots clés :

Information	Mot clé	Type de données
Position réelle cartésienne	DEF_Rlst	DOUBLE
Position de consigne cartésienne	DEF_RSol	DOUBLE

Information	Mot clé	Type de données
Position réelle spécifique à l'axe des axes de robot A1 ... A6	DEF_AIPos	DOUBLE
Position de consigne spécifique à l'axe des axes de robot A1 ... A6	DEF_ASPos	DOUBLE
Position réelle spécifique à l'axe des axes supplémentaires E1 ... E6	DEF_EIPos	DOUBLE
Position de consigne spécifique à l'axe des axes supplémentaires E1 ... E6	DEF_ESPos	DOUBLE
Courants moteur des axes de robot A1 ... A6	DEF_MACur	DOUBLE
Courants moteur des axes supplémentaires E1 ... E6	DEF_MECur	DOUBLE
Nombre de paquets de données arrivés trop tard	DEF_Delay	LONG
Paramètres technologiques dans le traitement principal (générateurs de fonctions 1 ... 6)	DEF_Tech.C1 ... DEF_Tech.C6	DOUBLE
Paramètres technologiques dans l'avance (générateurs de fonctions 1 ... 6)	DEF_Tech.T1 ... DEF_Tech.T6	DOUBLE

Exemple de fonction de lecture

- Structure XML configurée pour la transmission de données :

```
<SEND>
<ELEMENTS>
  <ELEMENT TAG="DEF_RIst" TYPE="DOUBLE" INDX="INTERNAL" />
  <ELEMENT TAG="DEF_AIPos" TYPE="DOUBLE" INDX="INTERNAL" />
  <ELEMENT TAG="DEF_MACur" TYPE="DOUBLE" INDX="INTERNAL" />
  <ELEMENT TAG="DEF_Delay" TYPE="LONG" INDX="INTERNAL" />
  <ELEMENT TAG="DEF_Tech.C1" TYPE="DOUBLE" INDX="INTERNAL" />
</ELEMENTS>
</SEND>
```

- Document XML transmis par le contrôleur du robot :

```
<Rob TYPE="KUKA">
  <RIst X="0.0" Y="0.0" Z="0.0" A="0.0" B="0.0" C="0.0" />
  <AIPos A1="0.0" A2="0.0" A3="0.0" A4="0.0" A5="0.0" A6="0.0" />
  <MACur A1="1.0" A2="1.0" A3="1.0" A4="1.0" A5="1.0" A6="1.0" />
  <Delay D="" />
  <Tech T11="0.0" T12="0.0" T13="0.0" T14="0.0" T15="0.0" T16="0.0"
    T17="0.0" T18="0.0" T19="0.0" T110="0.0" />
  <IPOC>123645634563</IPOC>
</Rob>
```



Le mot clé IPOC transmet un horodatage et est créé automatiquement.

7.4.6 Mots clés - Ecriture des données

Les mots clés sont des suites de lettres avec une signification constante attribuée. Ils ne peuvent pas être utilisés dans la structure XML au-delà de cette signification. L'écriture majuscule ou minuscule n'a pas d'importance. Un mot clé est valable en tant que mot clé quelle que soit l'écriture.

Mots clés

Les informations suivantes peuvent être écrites dans le contrôleur du robot par le biais de mots clés dans l'attribut TAG.

Information	Mot clé	Type de données
Paramètres technologiques dans le traitement principal (générateurs de fonctions 1 ... 6)	DEF_Tech.C1 ... DEF_Tech.C6	DOUBLE
Paramètres technologiques dans l'avance (générateurs de fonctions 1 ... 6)	DEF_Tech.T1 ... DEF_Tech.T6	DOUBLE

Mot clé dans l'attribut TAG pour sortie de messages sur le smartHMI :

Information	Mot clé	Type de données
Message de remarques ou d'erreurs	DEF_EStr	STRING

Types de messages

Les types de messages suivants peuvent surgir dans un document XML écrit et transmis par le système de capteurs :

- **<EStr> xxx </EStr>**: Message de remarque
- **<EStr>Error : xxx </EStr>** : Message d'acquittement (arrêt du robot)
- **<EStr/>** : aucun message en cas de Tag vide

Exemple de fonction d'écriture

- Structure XML configurée pour la réception de données :

```
<RECEIVE>
<ELEMENTS>
  <ELEMENT TAG="DEF_EStr" TYPE="STRING" INDX="INTERNAL" />
  <ELEMENT TAG="DEF_Tech.T2" TYPE="DOUBLE" INDX="INTERNAL" HOLDON="0" />
</ELEMENTS>
</RECEIVE>
```

- Document XML reçu par le système de robot :

```
<Sen Type="ImFree">
  <EStr>Message!</EStr>
  <Tech T21="0.0" T22="0.0" T23="0.0" T24="0.0" T25="0.0" T26="0.0"
    T27="0.0" T28="0.0" T29="0.0" T210="0.0" />
  <IPOC>123645634563</IPOC>
</Sen>
```



L'horodatage accompagné du mot clé IPOC est contrôlé. Le paquet de données n'est valide que si l'horodatage correspond à l'horodatage transmis auparavant.

8 Exemples

8.1 Configurations et de programmes à titre d'exemple



Les fichiers RSI, DIAGRAM et XML constituent une unité et doivent être transmis ensemble au contrôleur du robot.

Répertoire cible :

C:\KRC\Roboter\Config\User\Common\SensorInterface

Aperçu

RobotSensorInterface renferme une application à titre d'exemple au moyen de laquelle une communication Ethernet entre un programme de serveur et le contrôleur du robot peut être établie et testée. L'exemple d'application et d'autres exemples de configurations et de programmes figurent dans le répertoire DOC\Exemples du logiciel.

L'application de communication Ethernet donnée à titre d'exemple comprend les composants suivants :

Composants	Dossier
Programme de serveur TestServer.exe	...Ethernet\Server
Exemple de programme dans KRL : ■ RSI_Ethernet.src	...\Ethernet
Exemple de configuration du flux de signaux : ■ RSI_Ethernet.rsi ■ RSI_Ethernet.rsi.xml ■ RSI_Ethernet.rsi.diagram Fichier XML pour la liaison Ethernet : ■ RSI_EthernetConfig.xml	...\Ethernet\Config

Autres exemples de configurations et de programmes :

Composants	Dossier
Exemple de programme dans KRL : ■ RSI_CircleCorr.src	...\CircleCorr
Exemple de configuration du flux de signaux : ■ RSI_CircleCorr.rsi ■ RSI_CircleCorr.rsi.xml ■ RSI_CircleCorr.rsi.diagram	...\CircleCorr\Config
Exemple de programme dans KRL : ■ RSI_DistanceCtrl.src	...\DistanceCtrl
Exemple de configuration du flux de signaux : ■ RSI_DistanceCtrl.rsi ■ RSI_DistanceCtrl.rsi.xml ■ RSI_DistanceCtrl.rsi.diagram	...\DistanceCtrl\Config

Composants	Dossier
Exemple de programme dans KRL :	...\Transformations
<ul style="list-style-type: none"> ■ RSI_SigTransformation.src 	...\Transformations\Config
Exemple de configuration du flux de signaux :	
<ul style="list-style-type: none"> ■ RSI_SigTransformation.rsi ■ RSI_SigTransformation.rsi.xml ■ RSI_SigTransformation.rsi.diagram 	

8.1.1 Implémentation de l'application à titre d'exemple

Condition préalable

Système externe :

- Système d'exploitation Windows avec .NET-Framework 3.5 ou une version ultérieure

Contrôleur de robot :

- Groupe d'utilisateurs Expert
- Mode T1 ou T2.

Procédure

1. Copier le programme de serveur sur un système externe.
2. Copier les programmes KRL dans le répertoire C:\KRC\ROBOTER\KRC\R1\Program du contrôleur du robot.
3. Copier les exemples de configuration et le fichier XML pour la communication Ethernet dans le répertoire C:\KRC\ROBOTER\Config\User\Common\SensorInterface du contrôleur du robot.
4. Démarrer le programme de serveur sur le système externe.
5. Actionner le bouton de menu. La fenêtre **Server Properties** s'ouvre.
6. Les adresses IP disponibles du PC serveur sont affichées sous Available Network Interfaces.
7. Régler l'adresse IP de la connexion entre le robot et le PC dans le fichier XML de la connexion Ethernet.

8.1.2 Interface utilisateur du programme de serveur

Le programme de serveur permet de tester la connexion entre un système externe et le contrôleur du robot en établissant une communication stable vers le contrôleur du robot.

Pour ce faire, les données reçues sont analysées et l'horodatage courant du paquet est copié dans le document XML à envoyer. Le document XML peut être envoyé avec des données de correction ou des valeurs nulles.

Le programme de serveur contient les fonctionnalités suivantes :

- Envoi et réception de données dans le cycle du capteur
- Correction de mouvement libre, cartésienne via des éléments de commande
- Affichage des données reçues
- Affichage des données envoyées



Le programme de serveur test et le système d'exploitation Windows ne travaillent pas en temps réel. Il n'est pas possible de tirer des conclusions sur le comportement temporel ou la stabilité du système entier.

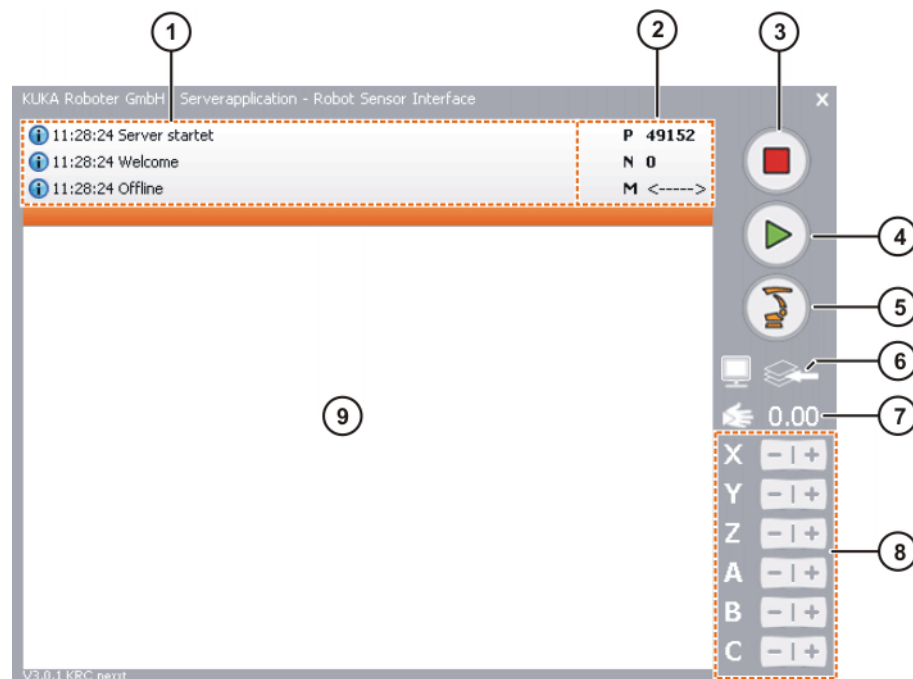


Fig. 8-1: Programme de serveur

Pos.	Description
1	Fenêtre de messages
2	<p>Affichage des paramètres de communication réglés</p> <ul style="list-style-type: none"> ■ P : numéro de port ■ N : index des cartes réseau ■ M : mode Communication <ul style="list-style-type: none"> ■ <----> : le serveur peut recevoir et transmettre des données. ■ <----- : le serveur peut seulement recevoir des données.
3	<p>Bouton Stop</p> <p>La communication avec le contrôleur du robot est terminée et le serveur est remis à zéro.</p>
4	<p>Bouton Start</p> <p>L'échange entre le programme de serveur et le contrôleur du robot est analysé. La première demande de connexion est accordée et utilisée en tant qu'adaptateur de communication.</p>
5	Bouton de menu permettant de régler les paramètres de communication
6	<p>Options d'affichage</p> <ul style="list-style-type: none"> ■ Flèche pointée vers la gauche : les données reçues sont affichées. (Par défaut). ■ Flèche pointée vers la droite : les données transmises sont affichées.
7	<p>Symbole de la main</p> <p>Un régulateur permet de régler la grandeur du pas de correction des mouvements par cycle de capteur.</p> <ul style="list-style-type: none"> ■ 0.00 ... 3.33

Pos.	Description
8	Boutons de correction de mouvement incrémentielle par cycle de capteur. La grandeur du pas est réglée par le biais du symbole de la main.
9	Fenêtre d'affichage Les données transmises ou reçues sont affichées en fonction de l'option d'affichage choisie. Les données affichées sont mises à jour dans le cycle du capteur.

8.1.3 Réglage des paramètres de communication dans le programme de serveur

Procédure

1. Cliquer sur le bouton Menu dans le programme de serveur.
La fenêtre **Server Properties** s'ouvre.
2. Régler les paramètres de communication.
3. Fermer la fenêtre.

Description

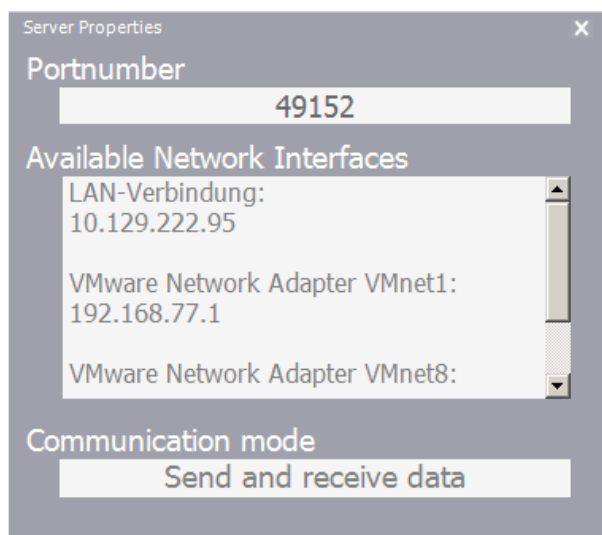


Fig. 8-2: FenêtreServer Properties

Elément	Description
Portnumber	Entrer le numéro de port de la connexion de socket. Le système externe attend la demande de connexion du contrôleur du robot sur ce port. Sélectionner un numéro libre n'étant pas occupé en service standard. Valeur par défaut : 49152
Available Network Interfaces	Affiche toutes les adresses IP disponibles, définies sur le PC utilisé.
Communication mode	Sélectionner le mode Communication. <ul style="list-style-type: none"> ■ Send and receive data : le serveur peut recevoir et transmettre des données. ■ Only receive data : le serveur peut seulement recevoir des données. Valeur par défaut : Send and receive data

8.1.4 Exemple de correction cartésienne via Ethernet

Le contrôleur du robot reçoit des données de correction cartésiennes d'un capteur et les transmet au robot. Le robot est uniquement déplacé selon la commande de correction sur la base de valeurs de correction relatives. Le système de coordonnées de référence pour l'activation de la correction est le système de coordonnées BASE.

Programme

```

1  DEF RSI_Ethernet( )
2  ; =====
3  ;
4  ; RSI EXAMPLE: ETHERNET communication
5  ; Realtime UDP data exchange with server application
6  ;
7  ; =====
8
9  ; Declaration of KRL variables
10 DECL INT ret; Return value for RSI commands
11 DECL INT CONTID; ContainerID
12
13 INI
14
15 ; Move to start position
16 PTP {A1 0, A2 -90, A3 90, A4 0, A5 90, A6 0}
17
18 ; Create RSI Context
19 ret = RSI_CREATE("RSI_Ethernet.rsi",CONTID,TRUE)
20 IF (ret <> RSIOK) THEN
21   HALT
22 ENDIF
23
24 ; Start RSI execution
25 ret = RSI_ON(#RELATIVE)
26 IF (ret <> RSIOK) THEN
27   HALT
28 ENDIF
29
30 ; Sensor guided movement
31 RSI_MOVECORR()
32
33 ; Turn off RSI
34 ret = RSI_OFF()
35 IF (ret <> RSIOK) THEN
36   HALT
37 ENDIF
38
39 PTP {A1 0, A2 -90, A3 90, A4 0, A5 90, A6 0}
40
41 END

```

Ligne	Description
16	Position de départ du mouvement guidé par capteur
19	RSI_CREATE() charge la configuration du flux de signaux dans un conteneur RSI.
25	RSI_ON() active le traitement du signal. ■ Mode de correction : Correction relative
31	RSI_MOVECORR() active le mouvement guidé par capteur.

Ligne	Description
34	RSI_OFF() désactive le traitement du signal.
39	Retour à la position de départ

Configuration du flux de signaux

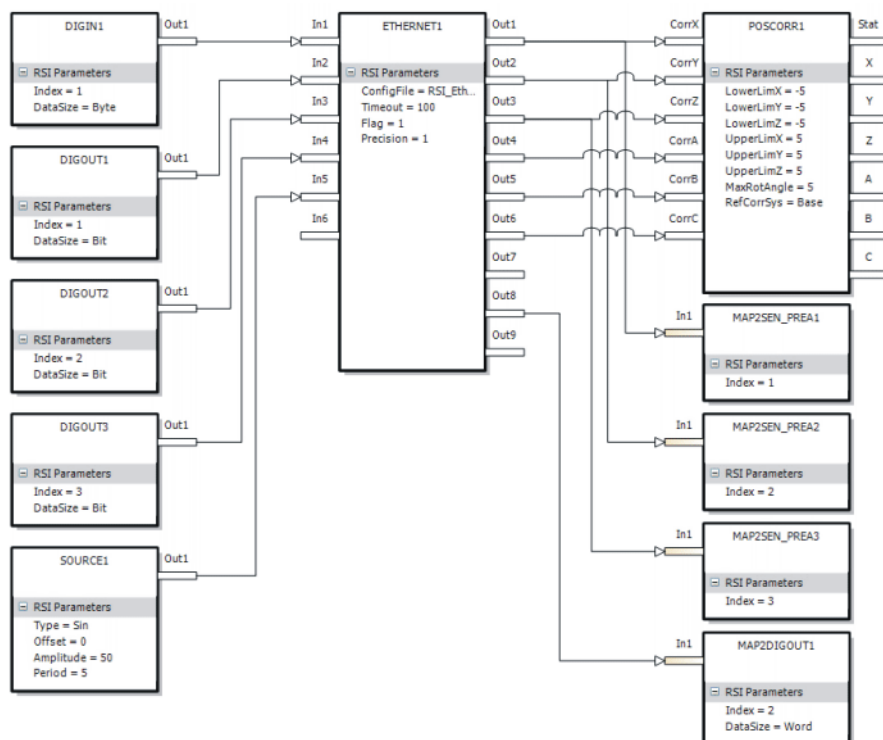


Fig. 8-3: Flux de signaux - Correction cartésienne via Ethernet

Objet RSI	Description
DIGIN1	Lit les données de capteurs via 8 entrées digitales et les transmet à l'interface Ethernet (entrée 1).
DIGOUT1 ... DIGOUT3	Lit les données du robot via 3 sorties digitales et les transmet à l'interface Ethernet (entrées 2 ... 4).
SOURCE1	Délivre périodiquement un signal sinusoïdal d'une amplitude de 50 toutes les 5 s.
ETHERNET1	Transmet les signaux entrants au système de capteurs via les entrées 2 ... 5 et reçoit les données de capteurs en retour via l'entrée 1. Les données de capteurs sont disponibles sur les sorties 1 ... 6 pour un traitement ultérieur.
POSCORR1	Lit les données de capteurs disponibles sur les entrées 1 ... 6 de l'interface Ethernet et détermine les données de correction cartésiennes.
MAP2SEN_PREA1 ... MAP2SEN_PREA3	Ecrit les données de correction cartésiennes dans la variable système \$SEN_PREA.
MAP2DIGOUT1	Accède aux signaux de traitement et active 16 sorties digitales.

8.1.5 Exemple de mouvement circulaire guidé par un capteur

Un mouvement circulaire guidé par capteur est configuré. Pour ce faire, un signal sinusoïdal est créé, lu en tant que sinus dans le sens Z, puis lu une nouvelle fois de manière temporisée dans l'objet de correction POSCORR en tant que sinus dans le sens Y. L'amplitude du signal est modifiée à posteriori dans le programme KRL à l'issue du premier passage du traitement des signaux.

Un mouvement circulaire de moindre importance est obtenu lorsque le traitement des signaux est redémarré à demi-amplitude.

Le robot est uniquement déplacé selon la commande de correction sur la base des valeurs de correction absolues dans le sens Y et Z. Le système de coordonnées de référence pour l'activation de la correction est le système de coordonnées BASE. Un Timer permet d'interrompre le mouvement exécuté par capteur à l'issue d'une durée définie.

Programme

```

1  DEF RSI_CircleCorr( )
2  ; =====
3  ;
4  ; RSI EXAMPLE: Lissajous circle
5  ; Create a circle movement with two sine corrections
6  ;
7  ; =====
8
9  ; Declaration of KRL variables
10 DECL INT ret; Return value for RSI commands
11 DECL INT CONTID; ContainerID
12
13 INI
14
15 ; Move to start position
16 PTP {A1 0, A2 -90, A3 90, A4 0, A5 90, A6 0}
17
18 ; Base in actual position
19 $BASE.X=$POS_ACT.X
20 $BASE.Y=$POS_ACT.Y
21 $BASE.Z=$POS_ACT.z
22
23 ; Create RSI Context
24 ret=RSI_CREATE("RSI_CircleCorr.rsi",CONTID)
25 IF (ret <> RSIOK) THEN
26   HALT
27 ENDIF
28
29 ; Start RSI execution
30 ret=RSI_ON(#ABSOLUTE)
31 IF (ret <> RSIOK) THEN
32   HALT
33 ENDIF
34
35 ; Sensor guided movement
36 RSI_MOVECORR()
37
38 ; Turn off RSI
39 ret=RSI_OFF()
40 IF (ret <> RSIOK) THEN
41   HALT
42 ENDIF
43
44 ; Modify RSI parameter
45 ret=RSI_GETPUBLICPAR(CONTID,"SOURCE1","Amplitude", fVar)
46 ...
49 ret=RSI_SETPUBLICPAR(CONTID,"SOURCE1","Amplitude", fVar/2)
50 ...
54 ; Start RSI execution
55 ret=RSI_ON(#ABSOLUTE)
56 ...
59

```

```

60 ; Sensor guided movement
61 RSI_MOVECORR()
62
63 ; Turn off RSI
64 ret=RSI_OFF()
65
66 ...
67
68
69 PTP {A1 0, A2 -90, A3 90, A4 0, A5 90, A6 0}
70
71 END

```

Ligne	Description
16	Point de départ du mouvement guidé par capteur
19 ... 21	Position du robot courante par rapport à la base.
24	RSI_CREATE() charge la configuration du flux de signaux dans un conteneur RSI.
30	RSI_ON() active le traitement du signal. ■ Mode de correction : Correction absolue
36	RSI_MOVECORR() active le mouvement guidé par capteur.
39	RSI_OFF() désactive le traitement du signal.
45	RSI_GETPUBLICPAR() lit l'amplitude courante réglée pour le signal (SOURCE1).
49	RSI_SETPUBLICPAR() assigne une nouvelle valeur à l'amplitude du signal (SOURCE1). L'amplitude est divisée par deux.
55	RSI_ON() active le traitement du signal. ■ Mode de correction : Correction absolue
61	RSI_MOVECORR() active le mouvement guidé par capteur.
64	RSI_OFF() désactive le traitement du signal.

Configuration du flux de signaux

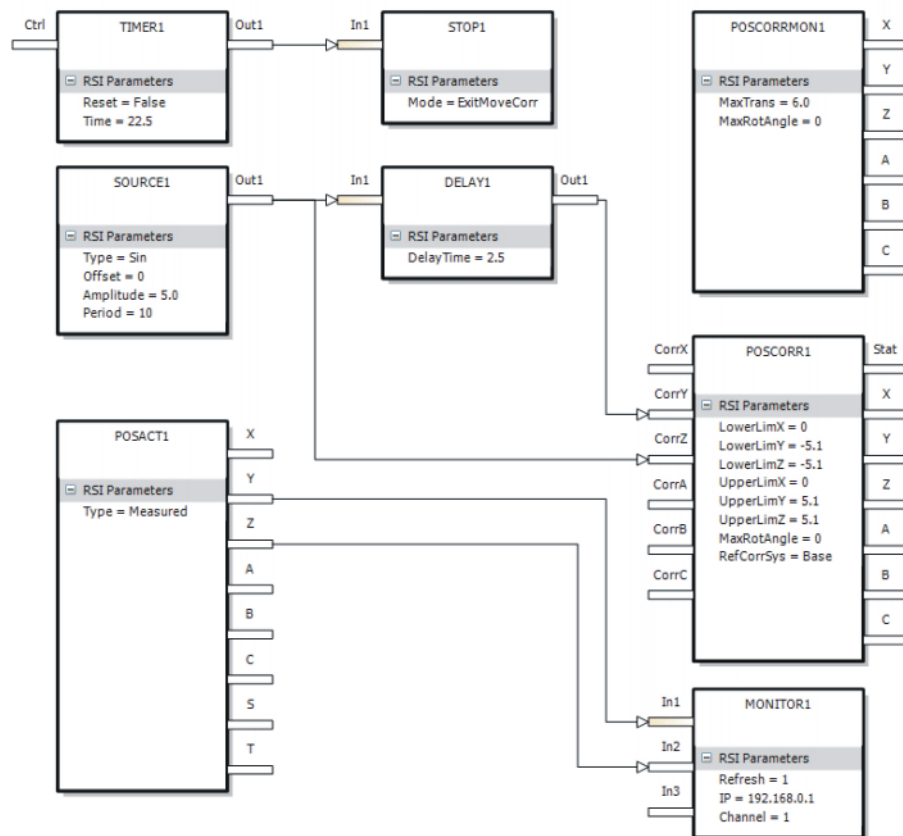


Fig. 8-4: Flux de signaux - Mouvement circulaire guidé par capteur

Objet RSI	Description
TIMER1	Le mouvement guidé par capteur est interrompu lorsque la durée réglée dans le Timer est écoulée.
STOP1	
POSCORRMON1	Limite la correction cartésienne totale maximale. ■ Déviation de translation maximale en X, Y, Z : 6 mm
SOURCE1	Délivre périodiquement un signal sinusoïdal d'une amplitude de 5.0 toutes les 10 s.
DELAY1	Le signal est retardé de 2.5 s.
POSCORR1	Lit la valeur de correction sinus dans le sens Z et la valeur de correction sinus retardée sans le sens Y.
POSACT1	Lit la position cartésienne réelle du robot dans le sens Y et Z. ■ Système de coordonnées de référence pour l'activation de la correction : BASE
MONITOR1	Les signaux suivants sont liés avec l'objet MONITOR et peuvent être affichés sur le contrôleur du robot avec le RSI Monitor : ■ Position réelle cartésienne du robot dans le sens Y et Z [mm]

8.1.6 Exemple de correction de trajectoire par rapport à la commande de distance

Une distance définie doit être maintenue par rapport à la pièce. Lorsqu'un traitement de signal est activé, un capteur mesure la distance par rapport à la pièce et avance de 100 mm dans le sens Y en effectuant un mouvement LIN. Parallèlement, une valeur de correction relative est déterminée et la trajectoire du mouvement LIN en Z est corrigée.

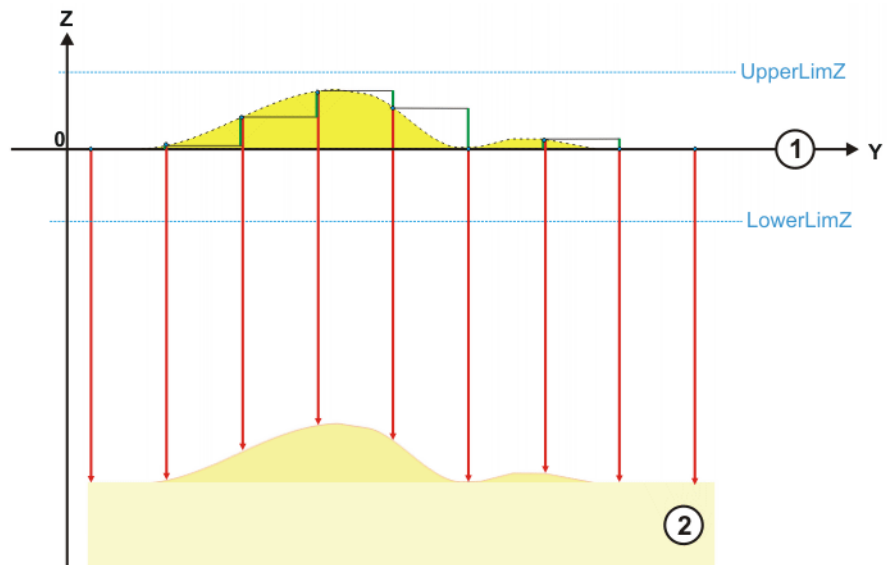


Fig. 8-5: Correction de trajectoire pour commande de la distance

1 Pièce

2 Trajectoire programmée

Programme

```

1  DEF RSI_DistanceCtrl( )
2  ; =====
3  ;
4  ; RSI EXAMPLE: Distance Ctrl
5  ; Move on a LIN path with superimposed corrections
6  ; Deviation from programmed path is controlled with
7  ; a analog input $ANIN[1]
8  ;
9  ; =====
10
11 ; Declaration of KRL variables
12 DECL INT ret; Return value for RSI commands
13 DECL INT CONTID; ContainerID
14
15 INI
16
17 ; Move to start position
18 PTP {A1 0, A2 -90, A3 90, A4 0, A5 90, A6 0}
19 $BASE=$POS_ACT
20
21 ; Create signal processing
22 ret=RSI_CREATE("RSI_DistanceCtrl.rsi")
23 IF (ret <> RSIOK) THEN
24   HALT
25 ENDIF
26
27 ; Start signal processing in relative correction mode
28 ret=RSI_ON(#RELATIVE)
29 IF (ret <> RSIOK) THEN
30   HALT
31 ENDIF
32
33 LIN_REL {Y 100}
34
35 ; Turn off RSI
36 ret=RSI_OFF()
37 IF (ret <> RSIOK) THEN

```

```

38   HALT
39   ENDIF
40
41   END

```

Ligne	Description
18	Point de départ de la correction de capteur
19	Position du système de coordonnées BASE dans le CDO courant
22	RSI_CREATE() charge la configuration du flux de signaux dans un conteneur RSI.
28	RSI_ON() active le traitement du signal. ■ Mode de correction : Correction relative
33	Déplacement LIN relatif dans le sens Y (100 mm)
36	RSI_OFF() désactive le traitement du signal.

Configuration du flux de signaux

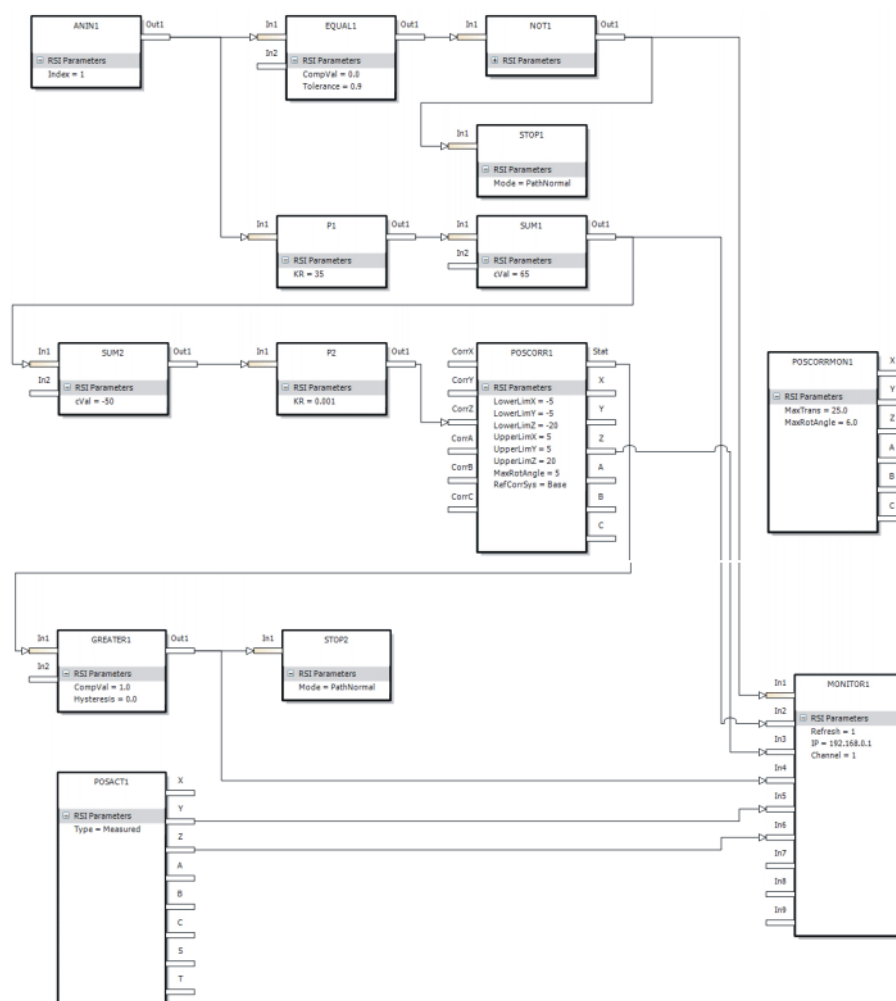


Fig. 8-6: Flux de signaux - Correction de trajectoire pour commande de la distance

Objet RSI	Description
ANIN1	Lecture du signal de capteur via une entrée analogique.
EQUAL1	EQUAL permet de contrôler si le signal de capteur se trouve dans une limite de tolérances. Si ce n'est pas le cas (NOT), le robot freine sur trajectoire.
NOT1	
STOP1	

Objet RSI	Description
P1	Le signal de capteur est converti avec P1, 5 V délivrent p. ex. une distance de 10 cm (= distance réelle). La distance réelle (SUM1) est additionnée à la distance de consigne (SUM2). Le résultat est la valeur de correction en cm dans le sens Z. P2 permet de convertir la valeur de correction en mm.
SUM1	
SUM2	
P2	
POSCORR1	<p>Lit, dans le sens Z, la valeur de correction disponible à la sortie de l'objet P2.</p> <ul style="list-style-type: none"> ■ Système de coordonnées de référence pour l'activation de la correction : BASE
POSCORRMON1	<p>Limite la correction cartésienne totale maximale.</p> <ul style="list-style-type: none"> ■ Déviation de translation maximale en X, Y, Z : 25 mm ■ Déviation de rotation maximale de l'angle de rotation : 6°
GREATER1	L'état de correction disponible à la sortie « stat » de l'objet de correction POSCORR est contrôlé. Si l'état de correction est >1, c.-à-d. que la correction admissible a été dépassée et limitée automatiquement à la correction maximale de ± 20 mm, le robot freine sur trajectoire.
STOP2	
POSACT1	Lit la position cartésienne réelle courante du robot dans le sens Y et Z.
MONITOR1	<p>Les signaux suivants sont liés avec l'objet MONITOR et peuvent être affichés sur le contrôleur du robot avec le RSI Monitor :</p> <ul style="list-style-type: none"> ■ Signal de capteur lu de manière analogique [V] ■ Distance réelle calculée [cm] ■ Valeur de correction dans le sens Z [mm] ■ Limitation de correction (true, false) ■ Position réelle cartésienne du robot dans le sens Y et Z [mm]

8.1.7 Exemple d'une transformation dans un nouveau système de coordonnées

La programmation d'une transformation de données de positions saisies par un capteur est ici décrite.

Un capteur qui saisit la position d'une pièce, une caméra p. ex., est monté à côté de l'outil sur la bride de montage du robot. Les données du capteur doivent être transformées dans le système de coordonnées BASE du contrôleur du robot à partir du système de coordonnées du capteur.

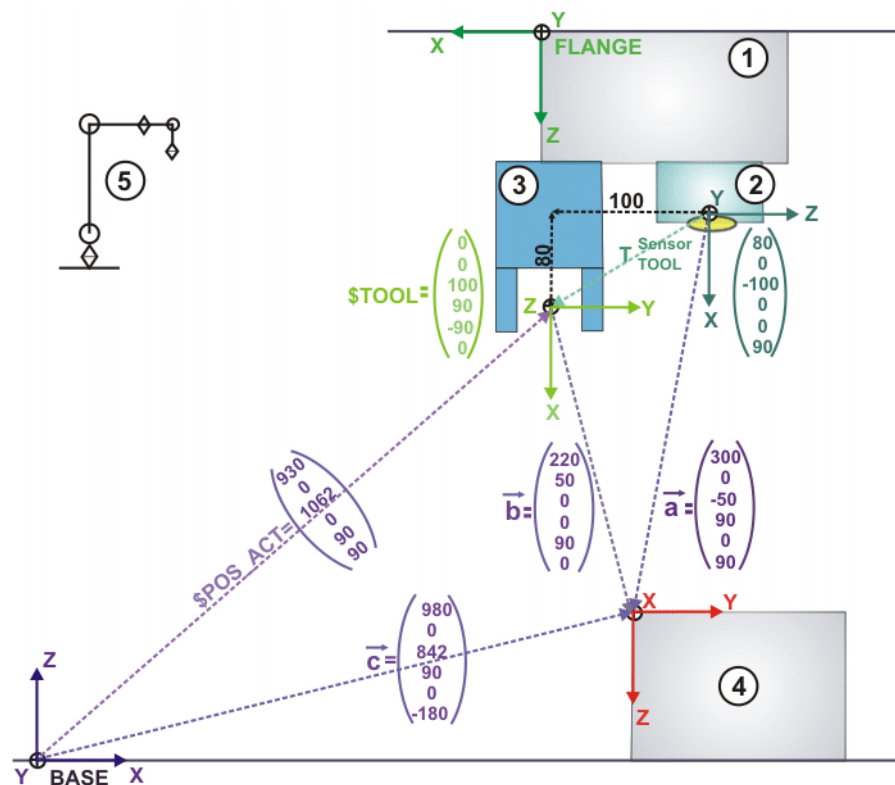


Fig. 8-7: Exemple d'une transformation

- | | | | |
|---|-------------------|---|-------------------|
| 1 | Bride de fixation | 4 | Pièce à usiner |
| 2 | Capteur | 5 | Position du robot |
| 3 | Outil | | |

Le capteur saisit la position et l'orientation d'une pièce dans le système de coordonnées du capteur (vecteur a). Le décalage et la rotation du capteur par rapport à l'outil ($T_{\text{capteur/outil}}$) sont indiqués dans l'objet RSI TRAFO_USERFRAME. TRAFO_USERFRAME transforme les données du capteur dans le système de coordonnées TOOL (vecteur b). L'objet RSI TRAFO_ROBFRAME est utilisé pour obtenir les données du capteur dans le système de coordonnées BASE. TRAFO_ROBFRAME transforme les données de l'outil dans le système de coordonnées BASE (vecteur c).

L'exemple numérique indiqué sur la figure peut être vérifié au moyen du programme exemple. Un KR 16 doit être défini pour ce faire. La position et l'orientation de la pièce dans le système de coordonnées BASE du contrôleur du robot (vecteur c) sont obtenues lorsque les signaux liés avec l'objet MONITOR sont affichés sur le contrôleur du robot avec RSI Monitor.

Programme

```

1  DEF RSI_SigTransformation( )
2  ; =====
3  ;
4  ; RSI EXAMPLE: Transformation of coordinates
5  ; Simulate a sensorsignal in relationship to
6  ; a flange mounted sensor. Transform the SIGNAL
7  ; to $BASE coordinates. Show the transformed
8  ; position in RSIMONITOR
9  ; =====
10
11 ; Declaration of KRL variables
12 DECL INT CONTID; ContainerID
13
14 INI

```

```

15
16 ; Move to start position
17 PTP {A1 0, A2 -90, A3 90, A4 0, A5 90, A6 0}
18 $TOOL = {X 0, Y 0, Z 100, A 90 ,B -90, C 0}
19 $BASE = $NULLFRAME
20
21 ; Create signal processing
22 IF (RSI_CREATE("RSI_SigTransformation.rsi") <> RSIOK) THEN
23     HALT
24 ENDIF
25
26 ; Start signal processing
27 IF (RSI_ON() <> RSIOK) THEN
28     HALT
29 ENDIF
30
31 wait sec 0.012
32
33 ; Turn off RSI
34 IF (RSI_OFF() <> RSIOK) THEN
35     HALT
36 ENDIF
37
38 END

```

Ligne	Description
17	Position de départ de la transformation
18	Position du système de coordonnées TOOL
19	Position du système de coordonnées BASE (NULLFRAME)
22	RSI_CREATE() charge la configuration du flux de signaux dans un conteneur RSI.
27	RSI_ON() active le traitement du signal. ■ Mode de correction : Correction relative
31	Les données de transformation sont calculées pendant le temps d'attente.
34	RSI_OFF() désactive le traitement du signal.

Configuration du flux de signaux

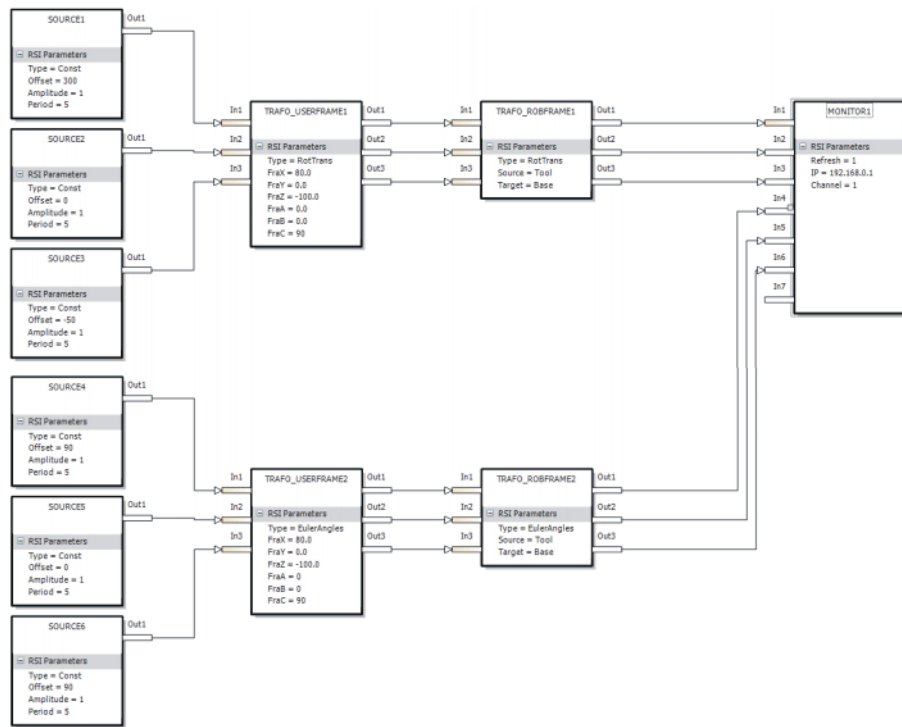


Fig. 8-8: Flux de signaux - Transformation

Objet RSI	Description
SOURCE1 ... SOURCE3	Délivre la position de la pièce dans le système de coordonnées du capteur (vecteur a) et transmet les données à TRAFO_USERFRAME1.
TRAFO_USERFRAME1	Transforme les données de position de la pièce dans le système de coordonnées du capteur dans le système de coordonnées TOOL du contrôleur du robot (vecteur b). Les données sont disponibles aux sorties de l'objet.
TRAFO_ROBFRAME1	Transforme les données de position de la pièce dans le système de coordonnées TOOL dans le système de coordonnées BASE du contrôleur du robot (vecteur c). Les données sont disponibles aux sorties de l'objet.
SOURCE4 ... SOURCE6	Délivre l'orientation de la pièce dans le système de coordonnées du capteur (vecteur a) et transmet les données à TRAFO_USERFRAME2.
TRAFO_USERFRAME2	Transforme l'angle d'orientation de la pièce dans le système de coordonnées du capteur dans le système de coordonnées TOOL du contrôleur du robot (vecteur b). Les données sont disponibles aux sorties de l'objet.
TRAFO_ROBFRAME2	Transforme l'angle d'orientation de la pièce dans le système de coordonnées TOOL dans le système de coordonnées BASE du contrôleur du robot (vecteur c). Les données sont disponibles aux sorties de l'objet.
MONITOR1	Les signaux suivants sont liés avec l'objet MONITOR et peuvent être affichés sur le contrôleur du robot avec le RSI Monitor : <ul style="list-style-type: none"> ■ Résultat de la transformation (vecteur c) : Position et orientation de la pièce dans le système de coordonnées BASE du contrôleur du robot

9 Diagnostic

9.1 Affichage de données de diagnostic concernant le RSI

- Procédure**
1. Dans le menu principal, sélectionner **Diagnostic > Ecran de diagnostic**.
 2. Dans le champ **Module**, sélectionner le module **Diagnostic RSI**.

Description Données de diagnostic concernant le RSI :

Nom	Description
Etat	Etat du traitement des signaux <ul style="list-style-type: none"> ■ Actif IPO : traitement des signaux en mode IPO ■ Actif IPO_FAST : traitement des signaux en mode IPO_FAST ■ Inactif : pas de traitement des signaux
Cycle de capteur	Durée de cycle du traitement des signaux
Compteur	Nombre de cycles de calcul depuis le début du traitement des signaux
Temps d'exécution	Durée nécessaire pour calculer le contexte RSI courant
Temps d'exécution (min.)	Durée de calcul minimale du contexte RSI courant
Temps d'exécution (max.)	Durée de calcul maximale du contexte RSI courant
Temps d'exécution (moyen)	Durée de calcul moyenne du contexte RSI courant
Objet Compteur	Nombre d'objets RSI créés
Mémoire	Mémoire totale disponible pour le RSI (octets)
Mémoire utilisée	Mémoire utilisée (octets)
Cycles de communication	Cycles de communication réussis depuis le départ du traitement des signaux
Perte totale	Nombre de pertes de paquets depuis le début du traitement des signaux
Qualité de la liaison	Qualité du traitement des signaux <ul style="list-style-type: none"> ■ 0 ... 100 % 100 % = tous les paquets sont arrivés correctement 0 % = aucun paquet n'est arrivé correctement.
Perte connexe maximale	Perte connexe de paquets la plus importante depuis le début du traitement des signaux

9.2 Protocole de défauts (table de messages)

Les messages d'erreurs de l'interface sont consignés par défaut dans un fichier LOG sous C:\KRC\ROBOTER\LOG\SensorInterface.

Le niveau LOG peut être modifié de manière à ce que des messages de remarques supplémentaires soient consignés.

9.2.1 Configuration du LOG-Level

Le niveau LOG peut être modifié dans le fichier C:\KRC\Roboter\Config\User\Common\Logging_RSI.xml.

Condition préalable

- Groupe d'utilisateur "Expert"
- Mode T1 ou T2
- Aucun programme n'est sélectionné.

Procédure

1. Ouvrir le fichier.
2. Modifier le niveau LOG dans cette ligne :

```
<Class Name="RSILogger1" LogLevel="error" />
```

3. Sauvegarder la modification.

Description

LogLevel :	Description
error	Les messages d'erreurs de l'interface sont protocolés.
infos	Les messages d'erreurs et les messages d'arrière plan de l'interface sont protocolés.

10 Messages

10.1 Messages en cours de fonctionnement

N°	Message	Cause	Remède
29000	<i>{Type} Correction générale autorisée dépassée: RSI est arrêté</i>	La température totale autorisée a été dépassée.	Acquitter le message
29001	<i>{Type} Correction hors de la plage autorisée: {Valeur}</i>	La correction commandée dépasse la plage admissible définie pour l'objet de correction.	<ul style="list-style-type: none"> ■ Augmenter la correction admissible ■ Contrôler le traitement du signal
29002	<i>Flux de signaux ({Mode}): l'objet {ObjName} provoque le défaut {Error-Code}</i>	L'objet RSI indiqué dans RSI-Visual ne peut pas être calculé.	Contrôler si l'objet RSI a été configuré correctement dans RSI-Visual, et si les fichiers de configuration nécessaires sont disponibles et corrects.
29004	<i>Défaut RSI interne</i>	Une valeur de retour inattendue a été délivrée par une fonction.	Si le problème persiste, contacter le service après-vente KUKA.
29005	<i>RSI ne peut activer aucune sortie à cause de la protection opérateur</i>	RSI ne peut activer aucune sortie en raison de la protection opérateur	Le signal de protection opérateur ne doit pas être activé.
29006	<i>RSI : dépassement du délai de calcul du signal {CalcTime} usec</i>	Le contexte RSI configuré est trop volumineux pour pouvoir être calculé dans les limites du temps disponible.	Réduire le contexte RSI

11 Annexe

11.1 Augmentation de la mémoire



Une augmentation de la mémoire est seulement possible après consultation de KUKA Roboter GmbH. (>>> 12 "SAV KUKA" Page 77)

Description

Si la mémoire disponible est insuffisante, il est conseillé de vérifier le mode de programmation dans KRL et la configuration du flux de signaux.

Condition préalable

- Niveau Windows

Procédure

1. Ouvrir le fichier C:\KRC\ROBOTER\Config\User\Common\RSI.XML.
2. Introduire la capacité de mémoire souhaitée dans la section <Interface>, dans l'élément <MemSize>.

```
<Interface>
  <MemSize>500000</MemSize>
</Interface>
```

3. Sauvegarder la modification et fermer le fichier.

11.2 Bibliothèque d'objets RSI

11.2.1 Objets RSI pour surveillance de la correction

Nom	Description
POSCORRMON	Limitation de la correction cartésienne globale Le programme du robot doit être réinitialisé en cas de dépassement. Les sorties de l'objet fournissent la correction globale courante.
AXISCORRMON	Limitation de la correction globale spécifique à l'axe Le programme du robot doit être réinitialisé en cas de dépassement. Les sorties de l'objet fournissent la correction globale courante.

11.2.2 Objets RSI pour transmission des signaux

Nom	Description
DIGIN	Délivre la valeur d'une plage d'entrées digitales \$IN.
DIGOUT	Délivre la valeur d'une plage de sorties digitales \$OUT.
ANIN	Délivre la valeur d'une entrée analogique \$ANIN.
ANOUT	Délivre la valeur d'une sortie analogique \$ANOUT.
SEN_PINT	Délivre la valeur des variables du système \$SEN_PINT.
SEN_PREA	Délivre la valeur des variables du système \$SEN_PREA.
POSACT	Délivre la position du robot cartésienne courante.
AXISACT	Délivre l'angle d'axe courant des axes de robot A1 ... A6.
AXISACTEXT	Délivre les positions courantes des axes supplémentaires E1 ... E6.
SOURCE	Générateur de signaux Génère un déroulement de signal défini, p. ex. via un signal constant, un signal de forme sinusoïdale ou cosinusoidale, etc.

Nom	Description
GEARTORQUE	Délivre les couples d'entraînement des axes de robot A1 ... A6.
GEARTORQUEEXT	Délivre les couples d'entraînement des axes supplémentaires E1 ... E6.
MOTORCURRENT	Délivre les courants moteur des axes de robot A1 ... A6.
MOTORCURRENTEXT	Délivre les courants moteur des axes supplémentaires E1 ... E6.
STATUS	Délivre les informations d'état du contrôleur du robot, p. ex. l'état courant de l'interprète Submit ou Robot, le mode de fonctionnement courant, etc.
OV_PRO	Délivre l'override de programme courant \$OV_PRO.

11.2.3 Objets RSI pour transformation des coordonnées

Nom	Description
TRAFO_USERFRAME	Transforme un vecteur des entrées 1 ... 3 en un nouveau système de coordonnées de référence avec un décalage et une rotation définis.
TRAFO_ROBFRAME	Transforme un vecteur des entrées 1 ... 3 d'un système de coordonnées de référence du robot dans un autre.

11.2.4 Objets RSI pour liaison logique

Nom	Description
AND	Liaison ET Il est possible de raccorder jusqu'à 10 signaux d'entrée.
OR	Liaison OU Il est possible de raccorder jusqu'à 10 signaux d'entrée.
XOR	Liaison OU-OU Il est possible de raccorder jusqu'à 10 signaux d'entrée.
NOT	Négation logique

11.2.5 Objets RSI pour liaison logique binaire

Nom	Description
BAND	Liaison ET binaire Lie l'entrée de signal 1 avec une valeur constante. Si plusieurs entrées de signaux sont liées, elles sont reliées entre elles. Il est possible de raccorder jusqu'à 10 signaux d'entrée.
BOR	Liaison OU binaire Lie l'entrée de signal 1 avec une valeur constante. Si plusieurs entrées de signaux sont liées, elles sont reliées entre elles. Il est possible de raccorder jusqu'à 10 signaux d'entrée.
BCOMPL	Complément binaire

11.2.6 Objets RSI pour comparaisons mathématiques

Nom	Description
EQUAL	Comparaison si égal Compare l'entrée de signal 1 avec une valeur constante, ou compare les entrées de signaux 1 et 2 entre elles.
GREATER	Comparaison si supérieur Compare l'entrée de signal 1 avec une valeur constante, ou compare les entrées de signaux 1 et 2 entre elles.
LESS	Comparaison si inférieur Compare l'entrée de signal 1 avec une valeur constante, ou compare les entrées de signaux 1 et 2 entre elles.

11.2.7 Objets RSI pour opérations mathématiques

Nom	Description
SUM	Addition de signaux Il est possible de raccorder jusqu'à 10 signaux d'entrée. Une valeur constante peut être additionnée avec le paramètre d'objet RSI cVal .
MULTI	Multiplication de signaux
ABS	Fonction Montant
POW	Fonction Puissance
MULTI	Fonction Sinus
COS	Fonction Cosinus
TAN	Fonction Tangente
ASIN	Fonction Arc sinus
ACOS	Fonction Arc cosinus
ATAN	Fonction Arc tangente
EXP	Fonction Exposant
LOG	Fonction Logarithme
CEIL	Plus petit nombre entier supérieur ou égal au signal d'entrée
FLOOR	Plus grand nombre entier supérieur ou égal au signal d'entrée
ROUND	Fonction Arrondi
ATAN2	Arc tangente du quotient des entrées 1 et 2. Le quadrant du résultat est calculé à partir du signe des signaux d'entrée.

11.2.8 Objets RSI pour commande des signaux

Nom	Description
P	Amplification du signal
PD	Objet différentiel proportionnel $y(k) = B0 * x(k) + B1 * x(k-1)$ $B0 = KR * (1 + (TV / <\text{cycle de capteur}>))$ $B1 = -KR * (TV / <\text{cycle de capteur}>)$

Nom	Description
I	Objet d'intégration (algorithme trapèze) $y(k) = B0 * (x(k) + x(k-1)) + y(k-1)$ $B0 = \text{<cycle de capteur>} / (2 * T1)$
D	Objet de différentiation $y(k) = B0 * (x(k) - x(k-1))$ $B0 = KD / \text{<cycle de capteur>}$
PID	Objet PID $y(k) = y(k-1) + B0 * x(k) + B1 * x(k-1) + B2 * x(k-2)$ $B0 = KR * (1 + TV / \text{<cycle de capteur>})$ $B1 = -KR * (1 - \text{<cycle de capteur>} / TN + 2 * TV / \text{<cycle de capteur>})$ $B2 = KR * TV / \text{<cycle de capteur>}$
PT1	Objet de temporisation de 1er ordre $y(k) = -A0 * y(k-1) + B0 * x(k)$ $A0 = -\exp(-\text{<cycle de capteur>} / T1)$ $B0 = KR * (1 - \exp(-\text{<cycle de capteur>} / T1))$
PT2	Objet de temporisation de 2e ordre $y(k) = -A0 * y(k-1) - A1 * y(k-2) + B0 * x(k) + B1 * x(k-1)$ Cas 1 : $T1 \neq T2$ <ul style="list-style-type: none"> ■ $Z1 = \exp(-\text{<cycle de capteur>} / T1)$ ■ $Z2 = \exp(-\text{<cycle de capteur>} / T2)$ ■ $A0 = -Z1 - Z2$ $A1 = Z1 * Z2$ ■ $B0 = (KP / (T1 - T2)) / (T1 * (1 - Z1) - T2 * (1 - Z2))$ ■ $B1 = (KP / (T1 - T2)) / (T2 * Z1 * (1 - Z2) - T1 * Z2 * (1 - Z1))$ Cas 2 : $T1 == T2$ <ul style="list-style-type: none"> ■ $Z0 = \exp(-\text{<cycle de capteur>} / T1)$ ■ $B0 = KP * (1 - Z0 * (\text{<cycle de capteur>} / T1 + 1))$ ■ $B1 = KP * Z0 * (Z0 + (\text{<cycle de capteur>} / T1) - 1)$
GENCTRL	Objet de traitement de signaux génériques jusqu'au 8e ordre $y(z) = B0*u(z) + B1*u(z-1) + B2*u(z-2) + \dots + B8*u(z-8) - A1*y(z-1) - A2*y(z-2) - \dots - A8*y(z-8)$
IIRFILTER	IIR-FILTER

11.2.9 Autres objets RSI

Nom	Description
TIMER	Lorsque la durée définie est écoulée, un flanc positif est sorti sur la sortie de signal « Out1 ».
LIMIT	Limite un signal aux valeurs situées dans une limite inférieure et supérieure (LowerLimit, UpperLimit).
MINMAX	Délivre le signal minimum et maximum courant sur tous les signaux d'entrée. Il est possible de raccorder jusqu'à 10 signaux d'entrée.
DELAY	Retarde le signal d'entrée pendant une durée déterminée.

Nom	Description
SIGNALSWITCH	Commutation entre deux chemins de signaux au moyen d'un signal de commande
ETHERNET	Communication Ethernet UDP en format de données XML Il est possible de définir jusqu'à 64 entrées et sorties dans le fichier de configuration. Les signaux sur les entrées sont transmis au partenaire de communication. Les données reçues par le partenaire de communication sont disponibles sur les sorties.

11.2.10 Objets RSI pour actions

Nom	Description
MAP2OV_PRO	Modifie l'override du programme (\$OV_PRO).
STOP	Arrêt d'un mouvement en cas de signal de flanc positif Un mouvement conduit uniquement par capteur avec RSI_MOVECORR peut être interrompu avec le mode ExitMoveCorr .
MAP2SEN_PINT	Modifie la valeur des variables du système \$SEN_PINT.
MAP2SEN_PREA	Modifie la valeur des variables du système \$SEN_PREA.
MAP2DIGOUT	Décrit une sortie digitale \$OUT ou une limite de sorties digitales.
MAP2ANOUT	Décrit une sortie analogique \$ANOUT.
SETDIGOUT	En cas de flanc positif, active une sortie digitale \$OUT. La sortie activée est aussi maintenue dans le cas d'un flanc négatif.
RESETDIGOUT	En cas de flanc positif, réinitialise une sortie digitale \$OUT. La sortie réinitialisée est aussi maintenue dans le cas d'un flanc négatif.
POSCORR	Activation de correction cartésienne avec limitation
AXISCORR	Activation de correction axe par axe avec limitation des axes de robot A1 à A6
AXISCORREXT	Activation de correction axe par axe avec limitation des axes supplémentaires E1 à E6
MONITOR	Ecran RSI Visualisation de jusqu'à 24 signaux RSI

12 SAV KUKA

12.1 Demande d'assistance

Introduction	Cette documentation comprenant des informations relatives au service et à la commande vous fera office d'aide lors de l'élimination de défauts. La filiale locale est à votre disposition pour toute autre demande.
Informations	<p>Pour traiter toute demande SAV, nous nécessitons les informations suivantes :</p> <ul style="list-style-type: none"> ■ Type et numéro de série du manipulateur ■ Type et numéro de série du contrôleur ■ Type et numéro de série de l'unité linéaire (si existante) ■ Type et numéro de série de l'alimentation en énergie (si existante) ■ Version du logiciel System Software ■ Logiciel en option ou modifications ■ Pack de diagnostic KrcDiag <p>En supplément pour KUKA Sunrise : projets existants, applications comprises</p> <p>Pour des versions de KUKA System Software antérieures à V8 : archives du logiciel (KrcDiag n'est pas encore disponible ici.)</p> ■ Application existante ■ Axes supplémentaires existants ■ Description du problème, durée et fréquence du défaut

12.2 Assistance client KUKA

Disponibilité	Notre assistance client KUKA est disponible dans de nombreux pays. Nous sommes à votre disposition pour toute question !
Argentine	<p>Ruben Costantini S.A. (agence) Luis Angel Huergo 13 20 Parque Industrial 2400 San Francisco (CBA) Argentine Tél. +54 3564 421033 Fax +54 3564 428877 ventas@costantini-sa.com</p>
Australie	<p>KUKA Robotics Australia Pty Ltd 45 Fennell Street Port Melbourne VIC 3207 Australie Tél. +61 3 9939 9656 info@kuka-robotics.com.au www.kuka-robotics.com.au</p>

Belgique	<p>KUKA Automatisering + Robots N.V. Centrum Zuid 1031 3530 Houthalen Belgique Tél. +32 11 516160 Fax +32 11 526794 info@kuka.be www.kuka.be</p>
Brésil	<p>KUKA Roboter do Brasil Ltda. Travessa Claudio Armando, nº 171 Bloco 5 - Galpões 51/52 Bairro Assunção CEP 09861-7630 São Bernardo do Campo - SP Brésil Tél. +55 11 4942-8299 Fax +55 11 2201-7883 info@kuka-roboter.com.br www.kuka-roboter.com.br</p>
Chili	<p>Robotec S.A. (agence) Santiago de Chile Chili Tél. +56 2 331-5951 Fax +56 2 331-5952 robotec@robotec.cl www.robotec.cl</p>
Chine	<p>KUKA Robotics Chine Co., Ltd. No. 889 Kungang Road Xiaokunshan Town Songjiang District 201614 Shanghai P. R. de Chine Tél. +86 21 5707 2688 Fax +86 21 5707 2603 info@kuka-robotics.cn www.kuka-robotics.com</p>
Allemagne	<p>KUKA Roboter GmbH Zugspitzstr. 140 86165 Augsburg Allemagne Tél. +49 821 797-4000 Fax +49 821 797-1616 info@kuka-roboter.de www.kuka-roboter.de</p>

France

KUKA Automatismes + Robotique SAS
 Techvallée
 6, Avenue du Parc
 91140 Villebon S/Yvette
 France
 Tél. +33 1 6931660-0
 Fax +33 1 6931660-1
commercial@kuka.fr
www.kuka.fr

Inde

KUKA Robotics India Pvt. Ltd.
 Office Number-7, German Centre,
 Level 12, Building No. - 9B
 DLF Cyber City Phase III
 122 002 Gurgaon
 Haryana
 Inde
 Tél. +91 124 4635774
 Fax +91 124 4635773
info@kuka.in
www.kuka.in

Italie

KUKA Roboter Italia S.p.A.
 Via Pavia 9/a - int.6
 10098 Rivoli (TO)
 Italie
 Tél. +39 011 959-5013
 Fax +39 011 959-5141
kuka@kuka.it
www.kuka.it

Japon

KUKA Robotics Japan K.K.
 YBP Technical Center
 134 Godo-cho, Hodogaya-ku
 Yokohama, Kanagawa
 240 0005
 Japon
 Tél. +81 45 744 7691
 Fax +81 45 744 7696
info@kuka.co.jp

Canada

KUKA Robotics Canada Ltd.
 6710 Maritz Drive - Unit 4
 Mississauga
 L5W 0A1
 Ontario
 Canada
 Tél. +1 905 670-8600
 Fax +1 905 670-8604
info@kukarobotics.com
www.kuka-robotics.com/canada

Corée	<p>KUKA Robotics Korea Co. Ltd. RIT Center 306, Gyeonggi Technopark 1271-11 Sa 3-dong, Sangnok-gu Ansan City, Gyeonggi Do 426-901 Corée Tél. +82 31 501-1451 Fax +82 31 501-1461 info@kukakorea.com</p>
Malaisie	<p>KUKA Robot Automation (M) Sdn Bhd South East Asia Regional Office No. 7, Jalan TPP 6/6 Taman Perindustrian Puchong 47100 Puchong Selangor Malaisie Tél. +60 (03) 8063-1792 Fax +60 (03) 8060-7386 info@kuka.com.my</p>
Mexique	<p>KUKA de México S. de R.L. de C.V. Progreso #8 Col. Centro Industrial Puente de Vigas Tlalnepantla de Baz 54020 Estado de México Mexique Tél. +52 55 5203-8407 Fax +52 55 5203-8148 info@kuka.com.mx www.kuka-robotics.com/mexico</p>
Norvège	<p>KUKA Sveiseanlegg + Roboter Sentrumsvegen 5 2867 Hov Norvège Tél. +47 61 18 91 30 Fax +47 61 18 62 00 info@kuka.no</p>
Autriche	<p>KUKA Roboter CEE GmbH Gruberstraße 2-4 4020 Linz Autriche Tél. +43 7 32 78 47 52 Fax +43 7 32 79 38 80 office@kuka-roboter.at www.kuka.at</p>

Pologne KUKA Roboter Austria GmbH
 Spółka z ograniczoną odpowiedzialnością
 Oddział w Polsce
 Ul. Porcelanowa 10
 40-246 Katowice
 Pologne
 Tél. +48 327 30 32 13 or -14
 Fax +48 327 30 32 26
 ServicePL@kuka-roboter.de

Portugal KUKA Sistemas de Automatización S.A.
 Rua do Alto da Guerra n° 50
 Armazém 04
 2910 011 Setúbal
 Portugal
 Tél. +351 265 729780
 Fax +351 265 729782
 kuka@mail.telepac.pt

Russie KUKA Robotics RUS
 Werbnaja ul. 8A
 107143 Moscou
 Russie
 Tél. +7 495 781-31-20
 Fax +7 495 781-31-19
 info@kuka-robotics.ru
 www.kuka-robotics.ru

Suède KUKA Svetsanläggningar + Robotar AB
 A. Odhners gata 15
 421 30 Västra Frölunda
 Suède
 Tél. +46 31 7266-200
 Fax +46 31 7266-201
 info@kuka.se

Suisse KUKA Roboter Schweiz AG
 Industriestr. 9
 5432 Neuenhof
 Suisse
 Tél. +41 44 74490-90
 Fax +41 44 74490-91
 info@kuka-roboter.ch
 www.kuka-roboter.ch

Espagne	<p>KUKA Robots IBÉRICA, S.A. Pol. Industrial Torrent de la Pastera Carrer del Bages s/n 08800 Vilanova i la Geltrú (Barcelona) Espagne Tél. +34 93 8142-353 Fax +34 93 8142-950 Comercial@kuka-e.com www.kuka-e.com</p>
Afrique du Sud	<p>Jendamark Automation LTD (agence)) 76a York Road North End 6000 Port Elizabeth Afrique du Sud Tél. +27 41 391 4700 Fax +27 41 373 3869 www.jendamark.co.za</p>
Taiwan	<p>KUKA Robot Automation Taiwan Co., Ltd. No. 249 Pujong Road Jungli City, Taoyuan County 320 Taïwan, R. O. C. Tél. +886 3 4331988 Fax +886 3 4331948 info@kuka.com.tw www.kuka.com.tw</p>
Thaïlande	<p>KUKA Robot Automation (M) Sdn Bhd Thailand Office c/o Maccall System Co. Ltd. 49/9-10 Soi Kingkaew 30 Kingkaew Road Tt. Rachatheva, A. Bangpli Samutprakarn 10540 Thaïlande Tél. +66 2 7502737 Fax +66 2 6612355 atika@ji-net.com www.kuka-roboter.de</p>
République Tchèque	<p>KUKA Roboter Austria GmbH Organisation Tschechien und Slowakei Sezemická 2757/2 193 00 Praha Horní Počernice République tchèque Tél. +420 22 62 12 27 2 Fax +420 22 62 12 27 0 support@kuka.cz</p>

Hongrie KUKA Robotics Hungaria Kft.
Fő út 140
2335 Taksony
Hongrie
Tél. +36 24 501609
Fax +36 24 477031
info@kuka-robotics.hu

Etats-Unis KUKA Robotics Corporation
51870 Shelby Parkway
Shelby Township
48315-1787
Michigan
Etats-Unis
Tél. +1 866 873-5852
Fax +1 866 329-5852
info@kukarobotics.com
www.kukarobotics.com

Royaume-Uni KUKA Automation + Robotics
Hereward Rise
Halesowen
B62 8AN
Royaume-Uni
Tél. +44 121 585-0800
Fax +44 121 585-0900
sales@kuka.co.uk

Index

A

ABS (objet RSI) 73
 ACOS (objet RSI) 73
 AND (objet RSI) 72
 ANIN (objet RSI) 71
 Annexe 71
 ANOUT (objet RSI) 71
 Aperçu, instructions RSI 35
 Aperçu, RobotSensorInterface 11
 Application à titre d'exemple, implémentation 52
 ASIN (objet RSI) 73
 Assistance client KUKA 77
 ATAN (objet RSI) 73
 ATAN2 (objet RSI) 73
 AXISACT (objet RSI) 71
 AXISACTEXT (objet RSI) 71
 AXISCORR (objet RSI) 16, 75
 AXISCORREXT (objet RSI) 16, 75
 AXISCORRMON (objet RSI) 71

B

BAND (objet RSI) 72
 BCOMPL (objet RSI) 72
 Bibliothèque d'objets RSI 8, 71
 BOR (objet RSI) 72
 Butées logicielles 21

C

Caractères 35
 CCS 8
 CEIL (objet RSI) 73
 Commande 29
 Communication 11
 Conditions requises par le système 23
 Configuration 27
 Configurations à titre d'exemple 51
 Connaissances nécessaires 7
 Connexion réseau 27
 Connexion réseau, configuration 27
 Conteneur RSI 8
 Contexte RSI 8
 Correction de capteur, mode de fonctionnement 15
 Correction de capteur, sécurité 21
 COS (objet RSI) 73
 Cycle de capteur 9

D

D (objet RSI) 74
 DELAY (objet RSI) 74
 Demande d'assistance 77
 Description du produit 11
 Diagnostic 67
 DIGIN (objet RSI) 71
 DIGOUT (objet RSI) 71
 Documentation, robot industriel 7
 Désinstallation, RobotSensorInterface 24
 Désinstallation, RSI Visual 25

E

Editeur de flux de signaux, ouverture 30
 Enregistrement des signaux, chargement sur l'écran RSI 34
 Enregistrement des signaux, sauvegarde 34
 EQUAL (objet RSI) 73
 Ethernet 8
 ETHERNET (objet RSI) 75
 Ethernet, interfaces 27
 Ethernet, réseau de capteurs 27
 Exemples 51
 EXP (objet RSI) 73

F

Fichier XML, liaison Ethernet 43
 FLOOR (objet RSI) 73
 Flux de signaux, chargement de la configuration 32
 Flux de signaux, liaison dans le programme KRL 42
 Flux de signaux, sauvegarde de la configuration 31
 Fonctionnement, traitement des signaux 12
 Fonctionnement, transmission des données 13
 Fonctions 11
 Formations 7
 Forme des signaux, affichage 33

G

GEARTORQUE (objet RSI) 72
 GEARTORQUEEXT (objet RSI) 72
 GENCTRL (objet RSI) 74
 GREATER (objet RSI) 73
 Groupe cible 7
 Générateur de fonctions 23, 28

I

I (objet RSI) 74
 ID de conteneur RSI 8
 IIRFILTER (objet RSI) 74
 Installation 23
 Installation, RobotSensorInterface 23
 Installation, RSI Visual 24
 Instructions RSI, aperçu 35
 Instructions RSI, comportement 40
 Interface utilisateur, RSI Monitor 32
 Interface utilisateur, RSI Visual 29
 Introduction 7
 IP 9
 IPO_FAST, mode Capteur 9, 12, 37
 IPO, mode Capteur 9, 18, 23, 28, 37
 IPO, mode capteur 12

K

KLI 9, 27
 KR C 9

L

LESS (objet RSI) 73
 Liaison Ethernet, fichier XML 43
 LIMIT (objet RSI) 74
 LOG (objet RSI) 73
 LOG-Level, configuration 68
 Logging_RSI.xml 68
 Logiciel 23

M

MAP2ANOUT (objet RSI) 75
 MAP2DIGOUT (objet RSI) 75
 MAP2OV_PRO (objet RSI) 75
 MAP2SEN_PINT (objet RSI) 75
 MAP2SEN_PREA (objet RSI) 75
 Marques 10
 Matériel 23
 Messages 69
 Messages d'erreurs 69
 MINMAX (objet RSI) 74
 Mise à jour, RobotSensorInterface 23
 Mode Capteur 9, 12, 36
 Mode Capteur, sécurité 21
 Mode de fonctionnement, correction de capteur 15
 Moniteur de diagnostic (option de menu) 67
 MONITOR (objet RSI) 75
 MOTORCURRENT (objet RSI) 72
 MOTORCURRENTTEXT (objet RSI) 72
 Mots clés, lecture des données 48
 Mots-clés, écriture des données 49
 MULTI (objet RSI) 73
 Mémoire, augmentation 71

N

NOT (objet RSI) 72
 Numéro de canal, réglage 32

O

Objet RSI 8
 OR (objet RSI) 72
 OV_PRO (objet RSI) 72

P

P (objet RSI) 73
 Paramètre d'objet RSI, activation 31
 Paramètre d'objet RSI, réglage 31
 Paramètres d'objet RSI 8
 Paramètres de flux de signaux, modification dans KRL 43
 PD (objet RSI) 73
 PID (objet RSI) 74
 POSACT (objet RSI) 71
 POSCORR (objet RSI) 16, 75
 POSCORRMON (objet RSI) 71
 POW (objet RSI) 73
 Programmation 35
 Programme de serveur 51
 Programme de serveur, interface utilisateur 52
 Programme de serveur, réglage des paramètres de communication 54

Programmes à titre d'exemple 51
 Propriétés des signaux, moniteur RSI 33
 Protocole de défauts 67
 PT1 (objet RSI) 74
 PT2 (objet RSI) 74

R

Remarques 7
 Remarques relatives à la sécurité 7
 RESETDIGOUT (objet RSI) 75
 RoboTeam 16, 23
 RobotSensorInterface, aperçu 11
 ROUND (objet RSI) 73
 RSI 8
 RSI Monitor 8
 RSI Monitor (option de menu). 32
 RSI Monitor, interface utilisateur 32
 RSI Visual 8
 RSI Visual, interface utilisateur 29
 RSI Visual, désinstallation 25
 RSI Visual, installation 24
 RSI_CHECKID() 39
 RSI_CREATE() 35
 RSI_DELETE() 36
 RSI_DISABLE() 40
 RSI_ENABLE() 40
 RSI_GETPUBLICPAR() 38
 RSI_MOVECORR() 38
 RSI_OFF() 37
 RSI_ON() 36
 RSI_RESET() 39
 RSI_SETPUBLICPAR() 38
 RSI.DAT 27
 RSIERRMSG 28
 RSITECHIDX 28

S

SAV, KUKA Roboter 77
 Schéma XML 48
 SEN_PINT (objet RSI) 71
 SEN_PREA (objet RSI) 71
 SETDIGOUT (objet RSI) 75
 SIGNALSWITCH (objet RSI) 75
 smarHMI 9
 SOURCE (objet RSI) 71
 STATUS (objet RSI) 72
 STOP (objet RSI) 75
 SUM (objet RSI) 73
 Sécurité 21

T

Table de messages 67
 TAN (objet RSI) 73
 Termes utilisés 8
 TestServer.exe 51
 TIMER (objet RSI) 74
 TRAFO_ROBFRAME (objet RSI) 72
 TRAFO_USERFRAME (objet RSI) 72
 Traitement des erreurs 28
 Traitement des signaux, fonctionnement 12
 Transmission des données, fonctionnement 13

TTS 9
Types 35

U
UDP 9

X
XML 9
XOR (objet RSI) 72

