
JAKA[®] | 节卡

JAKA ロボット

APP 取扱マニュアル



JAKA Zu[®] APP

JAKA® | 节卡

JAKA ロボット

APP 取扱マニュアル

JAKA Zu® APP

APP バージョン番号 : 1.5

注意：

協働ロボットの定義は国際 ISO 基準および他の国際基準に従っており、操作者の安全を保護するために、ロボットの本体を作業対象が人間である場合に応用することはお勧めしません。しかし、ロボットの作業対象が人間であることが避けられないとき、人員の安全を保護するために、ロボットの応用者や APP の開発者は、人員の安全が十分に保障されていることを確保してから、ロボットの本体に安全な、充分のテストされ、認証された安全保護システムを配置します。

この APP 取扱マニュアルに含まれた内容は、上海 JAKA ロボット科技有限公司（以下、JAKA と通称する）の専有財産であり、JAKA の同意書がなければ、あらゆる形式でその内容を使用することは禁じられています。

JAKA が定期的にこの APP 取扱マニュアルを修正と改善しているため、その内容は予告なく変更される場合があります。ご了承ください。本マニュアルをお使いになる前に、製品情報についてももう一度ご確認ください。

APP 取扱マニュアルに含まれた情報が JAKA の承諾ではないので、本マニュアルに含まれたいかなる誤りおよび本マニュアルで紹介された製品によって引き起こされた事故や間接損害に対しても、JAKA は一切の責任を負うものではありません。製品をお取り付け・ご利用になる前に、必ず本マニュアルをお読みください。

本マニュアルに含まれた写真はイメージです。実際とは異なる場合があります。

本体が改造、あるいは分解されたことのあるロボットに対して、JAKA はそのアフターサービスを行いません。

JAKA ロボットを使用・修理する際、必ず安全な設備を使用し、安全に関する条項に従ってください。

JAKA ロボットのプログラム設計者、ロボットシステムの設計とデバッグの関係者は、JAKA ロボットのプログラミング方法とシステムのインストールに精通している必要があります。

本マニュアルについて

このマニュアルには、JAKA ZU の APP のソフト概要、稼働環境、使用方法などの内容が含まれます。

このマニュアルは、JAKA ZU ソフトのインストール及び活用を図るために、ロボットの使用に関する基本的なトレーニングを受けたユーザーのみを対象としております。

その他

製品についてさらに詳細な情報は、右側のウェブサイト（www.jaka.com）の QR コードを読み取ってください。



目次

本マニュアルについて.....	3
第 1 章 はじめに.....	7
1.1 はじめに.....	7
1.2 ソフト機能.....	7
1.3 稼働環境.....	7
第 2 章 ソフトウェアの紹介.....	8
2.1 画面認識.....	8
2.1.1 画面レイアウト.....	8
2.1.2 画面ヘルプ.....	9
2.2 画面機能.....	9
2.2.1 コントローラを閉じる.....	9
2.2.2 ロボットログの読取仕方.....	10
2.2.3 信号の強さ.....	13
2.2.4 ロボットの接続.....	13
2.2.5 ロボットの起動/停止.....	15
2.2.6 ロボット状態モニタリング.....	16
2.2.7 バージョン情報の確認仕方.....	17
第 3 章 パラメーター設定.....	19
3.1 ロボットパラメーターの設定仕方.....	19
3.1.1 システム設定.....	19
3.1.2 操作設定.....	24
3.1.3 安全設定.....	31
3.1.4 プログラム設定.....	39
3.1.5 ハードウェアと通信.....	42
3.2 I/O 情報の構成仕方.....	54

3.2.1 コントローラ IO.....	55
3.2.2 ツール側 IO.....	62
3.2.3 機能 IO.....	65
3.2.4 I/O 増設拡張仕方.....	67
3.2.5 Modbus-TCP.....	67
3.2.6 Modbus-RTU.....	68
第 4 章 運動操作.....	69
4.1 運動制御.....	69
4.1.1 運動参考座標系の切替仕方.....	69
4.1.2 ユーザー座標系の切替.....	69
4.1.3 ツール座標系の切替.....	70
4.1.4 ロボット運動の精確制御仕方.....	70
4.1.5 運動スピードの制御.....	71
4.2 カルテシアン空間運動.....	71
4.3 関節運動.....	71
4.4 位置運動.....	72
第 5 章 スタートプログラミング.....	77
5.1 コマンド説明.....	77
5.1.1 プログラミングの新規作成.....	77
5.1.2 移動コマンド.....	77
5.1.3 IO コマンド.....	83
5.1.4 制御コマンド.....	84
5.1.5 計算コマンド.....	91
5.1.6 文字コマンド.....	93
5.1.7 通信コマンド.....	98
5.1.8 サブプログラムコマンド.....	101

5.1.9 変数コマンド.....	101
5.1.10 拡張コマンド.....	104
5.2 画面見直し.....	108
5.2.1 拡大.....	108
5.2.2 縮小.....	108
5.2.3 元に戻す.....	108
5.3 プログラム操作.....	109
5.3.1 プログラム実行.....	109
5.3.2 プログラム実行速度.....	109
5.3.3 プログラミングの新規作成.....	109
5.3.4 プログラムを保存する.....	109
5.3.5 プログラム名を付けて保存する.....	109
5.3.6 プログラムを開く.....	109
5.3.7 プログラムの高度操作.....	110
5.3.8 プログラムのシングルステップデバッグ.....	111
5.3.9 プログラムロック.....	111
5.4 プログラミングの仕方.....	112
5.4.1 プログラム的実行ロジック.....	112
5.4.2 コマンドの使い方.....	112
付録.....	113
付録一 ロボットパラメーターのデータ種類.....	113
付録二 力制御センサー I 型.....	114
付録三 力制御センサー II 型.....	117
付録四 力制御センサー III 型.....	123
付録五 ヴィジュアル保護システム.....	127
付録六 MODBUS ADDRESS シート.....	136

第 1 章 はじめに

1.1 はじめに

JAKA Zu は、ロボットのデモンストレーター機能を一つに集約した APP です。このソフトウェアは、ロボットの手動操作、プログラミング、パラメータ設定及び情報モニタリングなどの機能を持ち、協働ロボットの制御ソフトウェア（デモンストレーター）となっています。JAKA Zu は、かさばるハンドヘルドロボットティーチペンダントの代わりに、その機能を手軽な APP に統合しました。モバイル端末にインストールして、対象ロボットと同一ローカルネットワークに接続することで、伝統的なかさばるティーチペンダントや煩雑な電源コードなしに、軽々とロボット操作が実現できます。

1.2 ソフト機能

- 1) ユーザーによるロボットの一般情報とイベントログの確認が可能；
- 2) ユーザーによるロボットの手動操作が可能；
- 3) ユーザーによるロボットの I/O モジュールの設定と管理が可能；
- 4) ユーザーのプログラミングによるロボット操作が可能；
- 5) ユーザーによるロボットへのパラメーター設定が可能；

1.3 稼働環境

上手に使うためのヒント：タブレット端末は以下製品を推奨します：レノボ新ちゃん PAD

端末タイプ	タブレット	端末タイプ	コンピュータ
オペレーティングシステム	* Android™ 8.0 の端末をご利用ください。	オペレーティングシステム	Windows7 32/64bit 以上
プロセッサ	麒麟 695 またはスナップドラゴン 660 以上	プロセッサ	Intel Core I3 以上
メモリー容量	32GB	メモリー容量	32GB
システムメモリー	4GB	システムメモリー	4GB
スクリーン寸法	8.0 インチ以上	グラフィックカード	Intel HD Graphics 4000 以上
ネットワーク通信	Wi-Fi	ネットワーク通信	Wi-Fi または有線ネットワークカード

表 1-1 操作端末仕様表

第2章 ソフトウェアの紹介

2.1 画面認識

2.1.1 画面レイアウト

JAKA Zu ソフトメイン画面は主に機能バー、オンオフバーとメニューバーによって構成されます。図 2-1 に示した通り。

メニューバー：コントローラ管理、ロボット管理及びソフト管理によって構成されます。主な機能は APP 設定、ロボット本体とコントローラの設定及びロボット接続機能です。

オンオフバー：ロボット本体電源オンオフと起動オンオフによって構成されます。主な機能はロボット本体電源と起動のオンオフ操作です。

機能バー：プログラミング、運動制御、I/O 制御とモニタリング情報によって構成されます。主な機能はロボットプログラム管理、ロボット運動制御、コントローラ I/O 管理及びロボット状態モニタリングです。



図 2-1 JAKA Zu のメイン画面

2.1.2 画面ヘルプ

初めての使用は、APP のメイン画面の右上にある「ヘルプ」ボタンをクリックすると、JAKA Zu 機能について教えてくれます。^② ヘルプ画面は図 2-2 の通り。

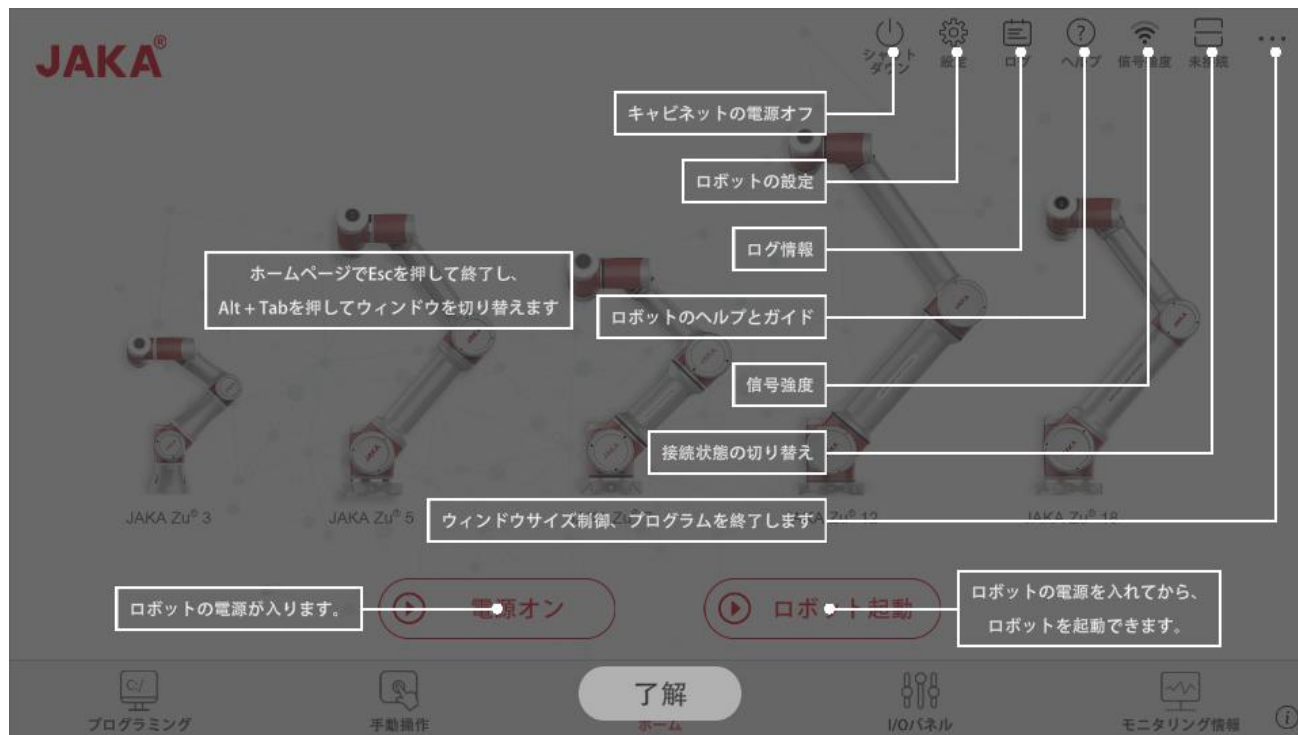


図 2-2 JAKA Zu のソフトヘルプ画面

2.2 画面機能

JAKA Zu は、JAKA 助ちゃんロボットティーチングソフトウェアで、ロボットパラメータの定義、ロボットのログ情報の読み、ロボットの接続と制御、ロボット本体状態の制御、ロボットのプログラミング制御、ロボットの手動制御、ロボットの I/O 信号の制御及びロボットの情報モニタリングなどを行うことができます。

2.2.1 コントローラを閉じる

メイン画面では、右上の「閉じる」ボタンをクリックすることで、コントローラの電源を切ることができます。^① 図 2-3 に示した通り。或いは外部コントローラのハンドルで停止ボタンを押して停止します。



図 2-3 コントローラのクローズ画面を閉じる

2.2.2 ロボットログの読取仕方

ログ情報は、ロボット操作イベントを記録する際に発生した関連情報の記録として、情報、警告とエラーに分かれます。ログ情報には、歴史データの処置、トラブルのトレース診断及び理解を深めるなど大きな役割があります。

ロボットを操作する際に関連なエラー警報が出ると、ログ情報のレビューによってエラー原因を調べたり、セルフチェックを始めることができます。解決不能なトラブルとなった場合、早急に問題解決ができるように、速やかに JAKA の技師にログ情報送信などを含めた連絡を取ってください。図 2-4 に示した通り、右上にある「ログ」ボタンをクリックしてログを確認します。



図 2-4 ログ情報画面

2.2.2.1 状態情報

ロボットの状態が変化したときに、この変化情報を保存され、ロボットの状態切替が成功したか否かや、一定期間に異常な状態切替えが起きたかどうかを確認することができます。状態情報ウィンドウは図 2-5 に示した通り：

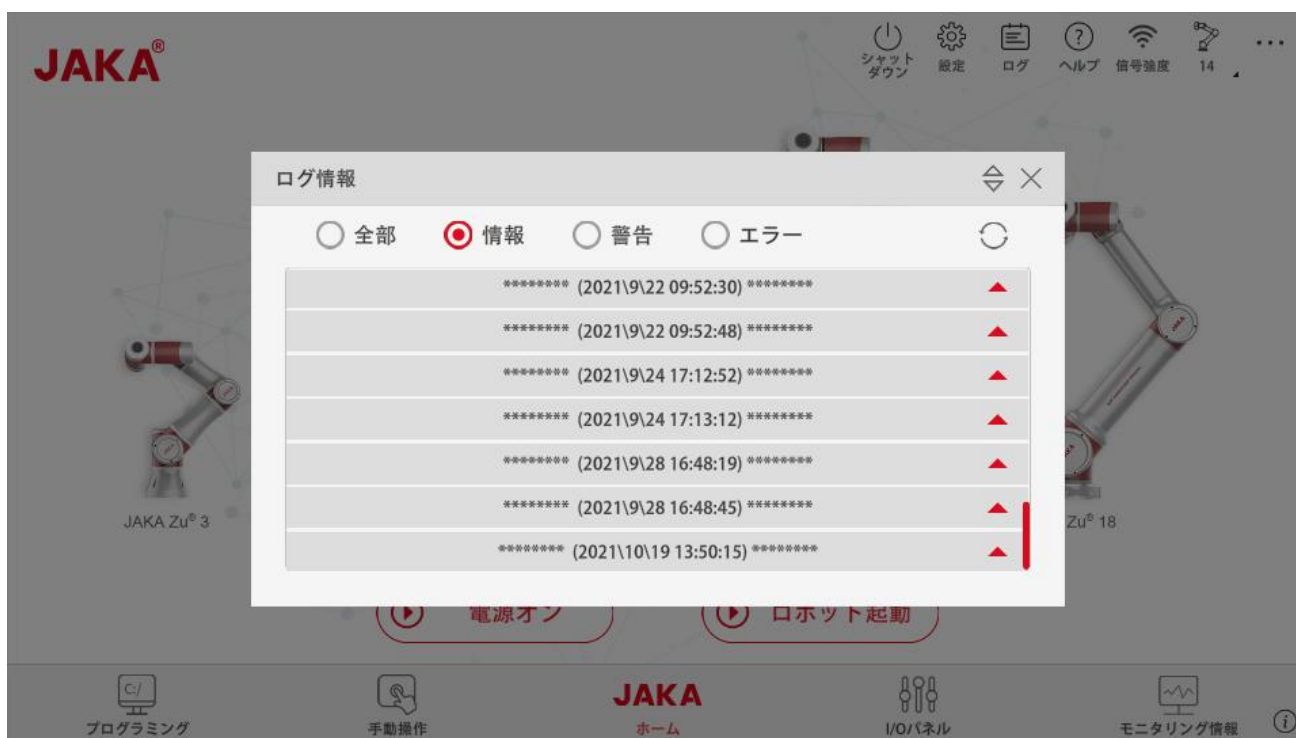


図 2-5 状態情報画面

2.2.2.2 警告情報

警告情報は、JAKA Zu ソフトウェアが異常な動作をしたときや、ロボットが異常な状態になったときに表示されるダイアログ情報です。この警告情報が事後のトラブルシューティングや異常モニタリングのためにリアルタイムにログ警告情報として保存されます。図 2-6 に示した通り。

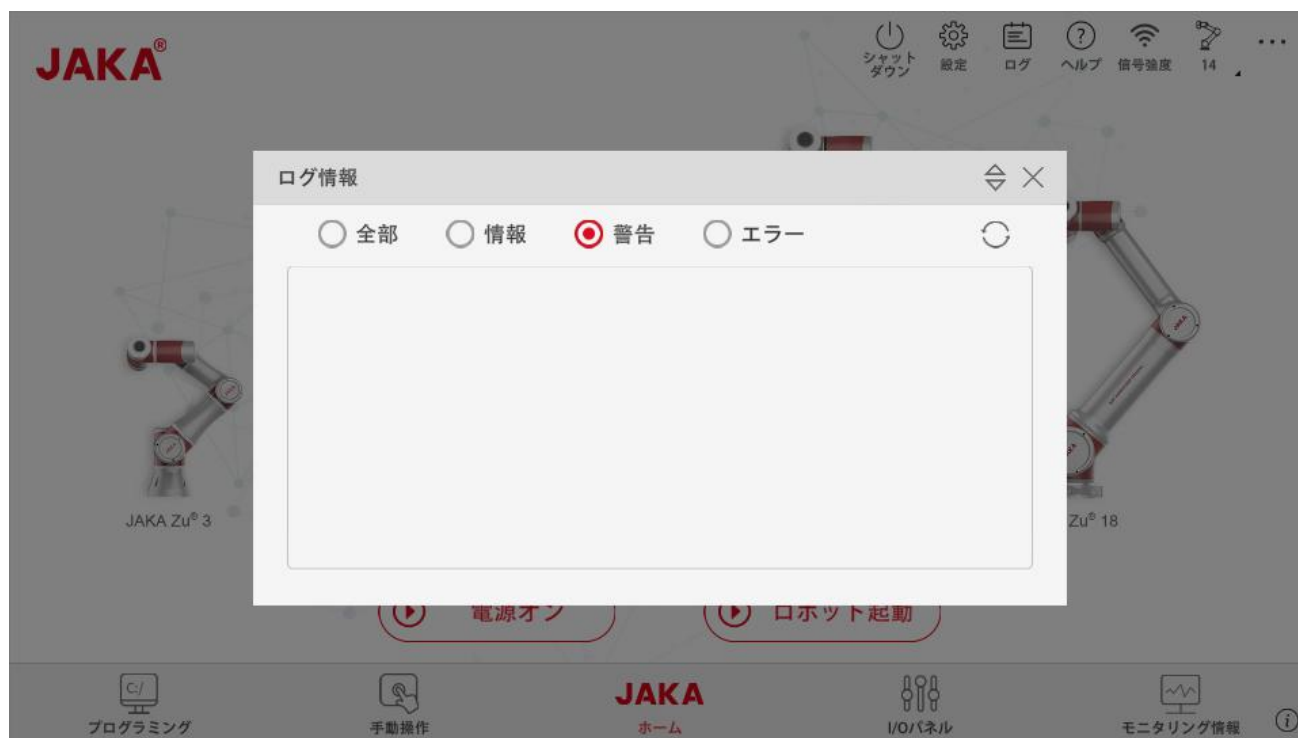


図 2-6 警告情報画面

2.2.2.3 エラー情報

エラー情報は、JAKA Zu ソフトウェアの使用中に誤った操作を行った場合や、ロボットにエラー状態が発生した場合に表示されるダイアログ情報です。JAKA Zu ソフトやロボット本体でエラー情報が発生した場合、トラブルシューティングや分析のためにログ情報として保存されます。図 2-7 に示した通り。



図 2-7 エラー情報画面

2.2.3 信号の強さ

JAKA Zuのメイン画面の右上には「信号の強さ」アイコンがあり、Wi-Fiによってロボットと接続した場合には、これを使ってネットワークの状態を把握できます。

2.2.4 ロボットの接続

JAKA Zu ソフトウェアは、1 つの APP で複数のロボットを管理することができます。JAKA Zu ソフトウェアをロボットと同じローカルネットワークに接続し、機能バーの「未接続」ボタンをクリックし、そして接続したいロボットをクリックすると、ログインウィンドウがダイアログしますので、暗証番号を入力してロボット接続をクリックします。

登録ウィンドウに下記三つの操作権限を選べます：操作員、技術員と管理員。

ロボットプログラムが停止状態で動作しているときにのみ、管理者や技術者がシステム設定、動作設定、安全設定、プログラム設定、ハードウェアと通信を編集・変更することができます。

操作員：「プログラムの実行/停止」、「ロボットの稼働/停止」、「ログ情報とロボット状態の表示」権限を持つ；デフォルト暗証番号：0

技術員：「プログラム編集」（及びプログラム編集に必要な関係設定など）及び操作員同等の権限を持つ；デフォルト暗証番号：0000

管理員：ソフトに関する全ての機能使用権限を持つ；デフォルト暗証番号：jakazuadmin

注：

1. プログラミング管理画面：操作員は、プログラムを開いて閲覧し、実行、一時停止、停止、プログラム速度の調整など、プログラムの実行と観察に必要な操作を行うことができますが、新規作成、編集、保存、名前を付けて保存、高度操作、シングルステップのデバッグ、プログラムロックなどは操作できません。
2. I/O パネル画面：操作員は、観察・閲覧、及びダイナミック I/O 編集と実行の状態切替のみ操作できます。
3. 設定画面：管理員はページ上のすべての内容を編集・変更・閲覧することができ、操作員はユーザー管理を

除くすべてのページを閲覧することができ、技術者はユーザー管理を除くすべてのページを閲覧し、セキュリティ設定を除く内容を変更することができます。図 2-8 に示した通り。

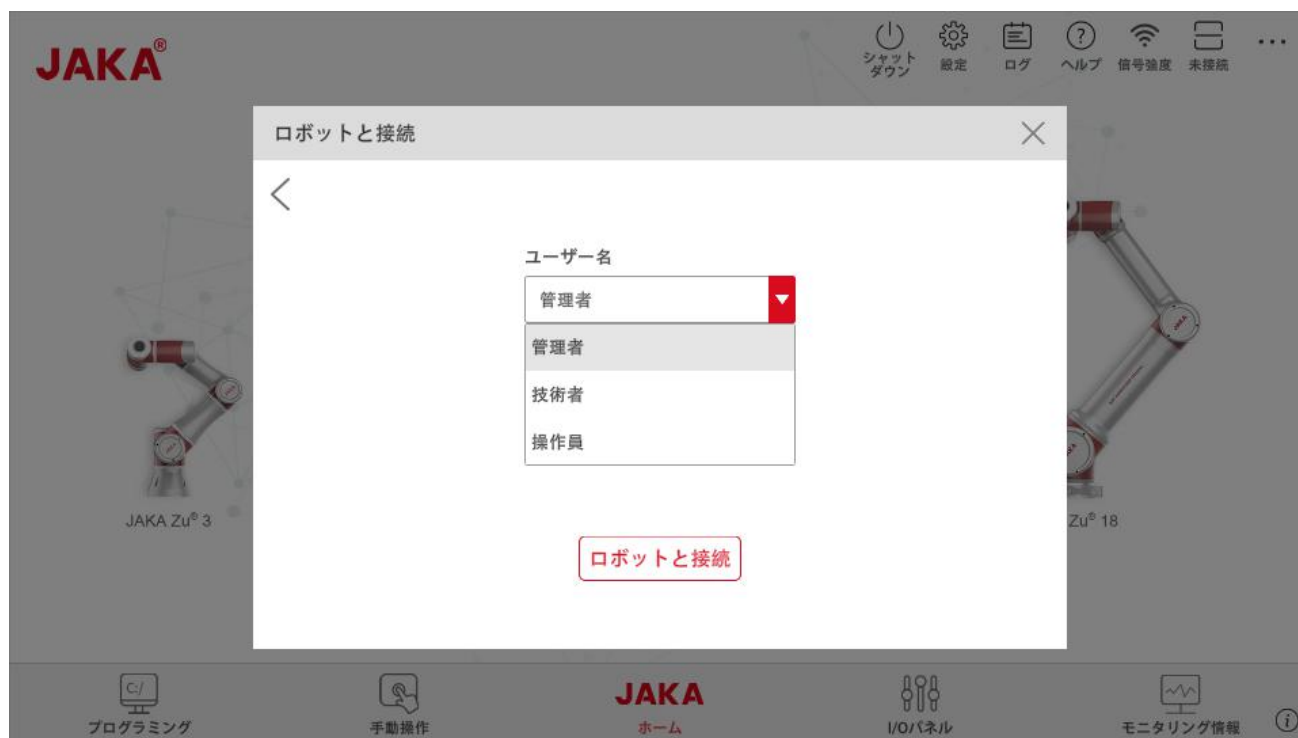


図 2-8 権限登録画面

2.2.4.1 ロボットへの登録仕方

まず、JAKA Zu ソフトをロボットと同じローカルネットワークに接続し、ソフトウェアのメイン画面に入ってから、右上の「未接続」ボタンをクリックすると、ロボット接続画面に入ります。この画面では、現在ローカルネットワーク上にある全てのロボット情報（ロボット ID、名前、IP アドレス、コントローラのバージョンと状態情報）を確認することができるので、接続したいロボットをクリックすると、登録画面に移ります。そこから登録権限を選んで暗証番号を入力してから、ロボット接続をクリックします。操作手順は図 2-9 に示した通り：



図 2-9 登録画面

2.2.4.2 接続情報の確認仕方

現在接続されているロボットの情報を確認する場合は、右上の「ロボット」アイコンをクリックし、ダイアログウィンドウの下部にある「拡大」ボタンをクリックすると、ロボット接続画面が表示されます。そこから現在接続されているロボットや、現在 LAN 上のロボットの情報を見ることができます。下図 2-10 に示した通り。

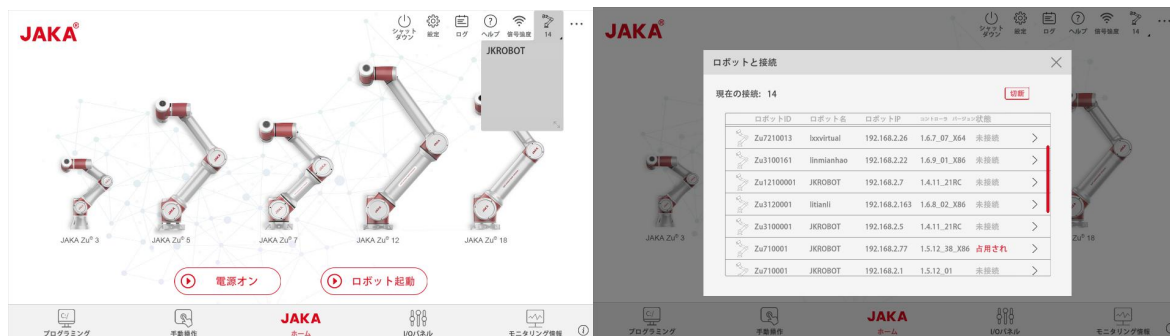


図 2-10 ロボットの接続画面

2.2.5 ロボットの起動/停止

ロボットを起動する前に、コントローラを起動する必要があります。ハンドルオンオフボタンを 1～2 秒長押ししてコントローラを起動し、正常起動完了（ハンドルの JAKA 点滅ランプが青くなる）を確認してから、ロボット本体を接続し、電源を入れて起動してください。

図 2-11.1 に示した通り、まず「本体電源を入れる」をクリックして、ソフトウェアメイン画面上のロボット端末指示ランプが青くなる、或いはロボット本体の端末指示ランプが青くなると、ロボットの電源が正常に入れたこととなります。再び「ロボットを起動する」をクリックして、ソフトウェアメイン画面上のロボット端末指示ランプが緑になる、或いはロボット本体の端

末指示ランプが緑になると、ロボット本体の起動が成功し、ロボットが正常に起動します。図 2-11.2 に示した通り。

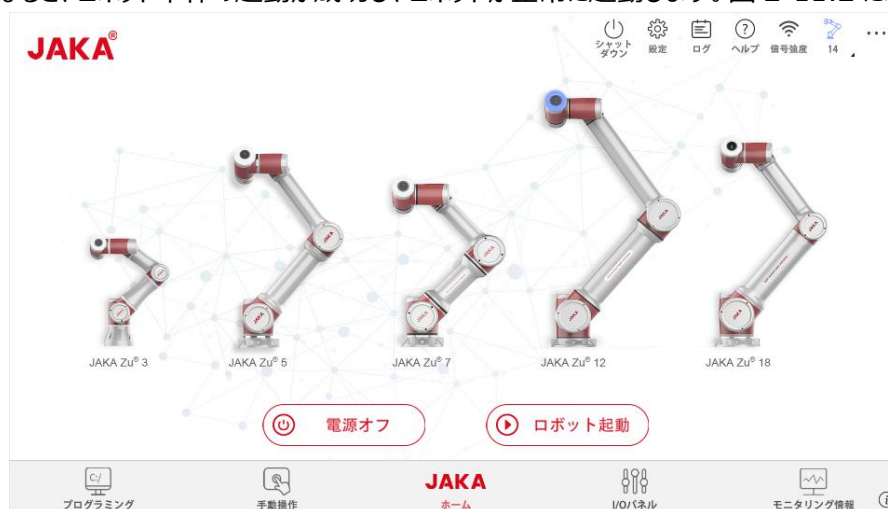


図 2-11.1 ロボットオンオフ制御画面

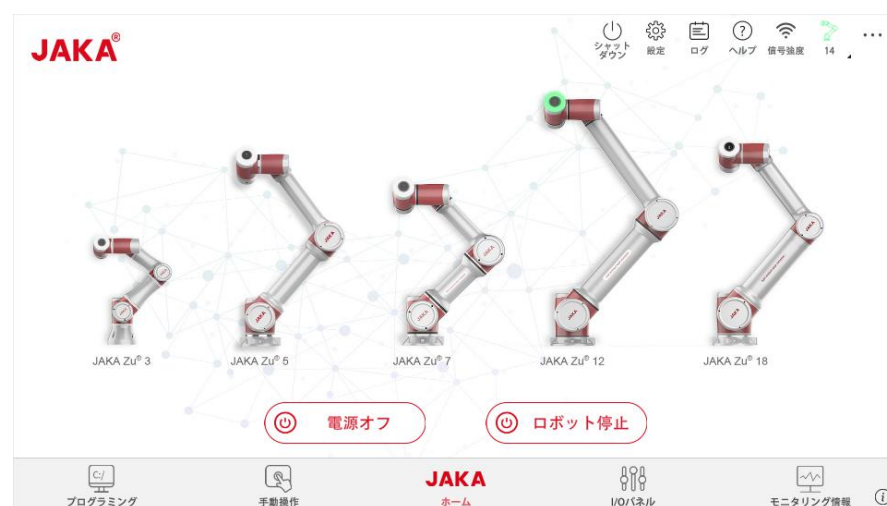


図 2-11.2 ロボットオンオフ制御画面

ロボットを停止させる手順は、まず「ロボット停止」をクリックします。ソフトウェアのメイン画面上のロボット端末指示ランプが青くなるか、ロボット本体の端末指示ランプが青くなると、ロボット本体の停止が成功したことになる。そして、再び「本体電源を切る」をクリックします、ソフトウェアのメイン画面上のロボット端末指示ランプが無色となるか、ロボット本体の端末指示ランプが消えると、ロボットの電源が正常にオフとなり、ロボット停止に成功します。

コントローラの停止方法は、ハンドルオンオフボタンを 3~5s 長押しして、コントローラを停止させ、ハンドル上にある JAKA 点滅ランプが消えると、コントローラの停止に成功します。

注：

- 1、ロボット本体を起動する前は、まずコントローラを起動すること。
- 2、ロボット本体を起動する前は、まずロボット本体の電源を入れること。
- 3、ロボット本体の電源を切る前は、まずロボット本体を停止させること。
- 4、ロボットの電源を切らず停止せずにしてコントローラの電源を切れたり、ロボット稼働中に電源を切ったりすると、未知の障害が発生する可能性があるため厳禁です。

2.2.6 ロボット状態モニタリング

JAKA Zu ソフトでは、ロボットの情報をモニターすることができます。ロボットやコントローラ、関節の情報を見たいとき

には、ソフトウェアのトップページからモニター情報画面にアクセスします。コントローラの情報に基づくコントローラの温度、ロボット瞬时パワー及びロボットの電流値をモニタリングできます。

関節情報に基づく関節一から関節六までの電流、電圧と温度情報をモニタリングできます。図 2-12 に示した通り。

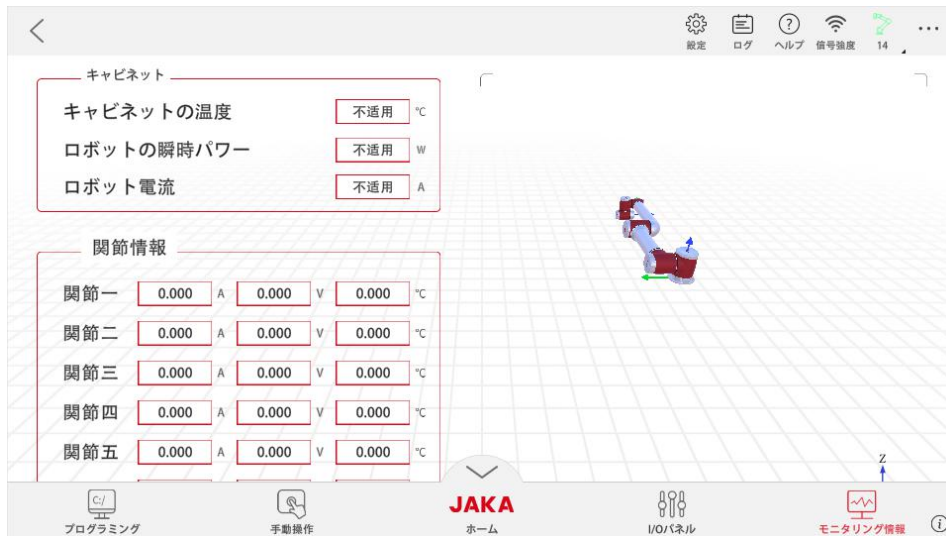


図 2-12 モニタリング情報画面

2.2.7 バージョン情報の確認仕方

JAKA Zu ソフトバージョン、コントローラバージョンとサーボバージョンを互いに一致させる必要がある。異なるバージョンを用いると、ロボットとの接続不可になりやすい。バージョンの不一致が出た場合、そのバージョン情報を関係 JAKA スタッフに連絡し、正しいバージョンを受け取ってから接続制御を行うこと。図 2-13 に示した通り。



図 2-13 バージョン画面

2.2.7.1 バージョン情報

JAKA Zu ソフトウェアを開き、ロボットを接続し、ソフトウェアのメイン画面の右下にある「詳しく見る」ボタンをクリックすると、バージョン情報ウィンドウが表示され、APPのバージョン、サーボのバージョン、コントローラのバージョン、ロボット番号を確認することができます。①

注：サーボバージョンはロボット本体の電源を入れてから確認できます。

2.2.7.2 バージョン言語

JAKA Zu ソフトは中国語、英語と日本語の 3 言語切替可能で、ソフトを開きメイン画面に入り、右下にある「詳しく見る」ボタンをクリックして、言語設定にて必要な言語バージョンの切替ができます。① 自動に設定された場合、APP は自動的に現在機器のシステム言語を認識して、ソフトの言語切替を行います。

第 3 章 パラメーター設定

APP 画面の右上にある設定ボタンをクリックして設定画面に入ります。設定画面には各々システム設定、操作設定、セキュリティ設定、プログラム設定及びハードウェアと通信という五つのカテゴリーに分かれます。

3.1 ロボットパラメーターの設定仕方

3.1.1 システム設定

3.1.1.1 時間と名前の変更

初期設定はロボット名前とシステム時間に対する設定となります。図 3-1 に示した通り。



図 3-1 初期設定画面

(1) ロボット名前の設定

目的：ロボット名前を変更すると、ロボット接続画面にて、そのロボット名前を用いていち早く接続したいロボットを見つけ出すことができます。ロボットに接続してからも、APP 右上の情報から、いち早く現在接続中のロボットを見つけ出すことができます。

(2) 時間設定

目的：システム時間をコントローラ内の時間とユーザーのローカル時間に一致させるように変更します。

3.1.1.2 ロボット IP アドレスの変更

この画面にて、ロボットによる IP アドレス取得方式を選ぶことができます。

ロボットの出荷デフォルトは自動 IP アドレス取得となっています。ロボットの IP を定着させる場合には、これから続く全てのローカルネットワーク経由ロボットとの通信装置（デバッグ用 Pad も含む）を、当該ロボットと同一サブネットに配属させる必要があります。図 3-2 に示した通り。

注：

1. ロボット IP アドレス定着させる設定は、ロボットが電源落ち且つ停止状態でのみ操作できます。

2. V2.1 版コントローラ：①制御盤パネルのネットワークポートは、出荷デフォルト設定として 10.5.5.x セグメントになっています。このネットワークポートを使用する場合は、接続機器の IP アドレスが 10.5.5.101～10.5.5.254 の範囲内になるように設定するか、機器を IP アドレスのダイナミック取得となるように設定する必要があります。②制御盤底部のネットワークポートは、出荷デフォルトとして IP アドレスダイナミック取得が設定されています。二つのネットワークポートの IP

セグメントが同一になってはならないことに注意します。

3. MiniCab版コントローラ：①LAN1ネットワークポートの出荷デフォルト設定は10.5.5.xセグメントである。LAN1ネットワークポートに接続する場合、導入機器のIPアドレスが10.5.5.101～10.5.5.254の範囲内になるように、或いは導入機器をIPアドレスのダイナミック取得に設定する必要があります。②LAN2ネットワークポートの出荷デフォルト設定はIPアドレスのダイナミック取得となっています。二つのネットワークポートのIPセグメントが同一になってはならないことに注意します。



図 3-2 ネットワーク設定画面

3.1.1.3 追加プログラム

JAKA Zu ソフトウェアでは、特定のユーザー向けに機能を追加開発する「AddOn」機能を使った二次開発をサポートしています。ユーザーが特定の二次開発機能の使用を希望する場合は、JAKA の関連技術スタッフに連絡することができます。

右上のプラス記号をクリックし、ファイルのローカル所在位置に基づいた追加プログラムパッケージを選択し、「確認」をクリックします。図 3-3 に示した通り。



図 3-3 追加プログラム管理画面

3.1.1.4 ファームウェアバージョンのバージョンアップ

ユーザーは、ファームウェアのバージョンアップ画面で、ロボットの SCB とサーボバージョンをバージョンアップできます。図 3-4 と 3-5 に示した通り

正しいバージョンアップの手順は：コントローラのアップグレード→APP のアップグレード→SCB アップグレード→サーボのアップグレード→CAN カードのアップグレードとなりますが、その内数項目のみを対象としても、上記の手順に従ってください。

SCB のアップグレード手順：

- 1) SCB アップグレードパッチを機器ローカルファイルにダウンロードします。
- 2) APP を開いてロボットに接続し、ロボットの電源がオフの状態にしておきます。
- 3) APP メイン画面の右上にある設定ボタンをクリックして、「システム設定」→「ファームウェアアップグレード」画面に入り、「ファイル選択」の空白部分をクリックして、SCB アップグレードパッチファイル（パッチファイルの名前は変更せず、元の状態にしておくこと）を見つけてアップロードしてください。
- 4) コントローラの電源が自動的に切れるのを待って、ハンドルを使ってコントローラの電源を入れます（この時、SCB がアップグレード中なので、ハンドルは無電源の状態、ステータスランプも点灯しません）。
- 5) 完全に立ち上げるのを待って（APP 接続画面にてロボット検出完了な状態）、APP を用いて再度ロボットに接続します。
- 6) 第（3）ステップを繰り返します。
- 7) コントローラの電源が自動的に切れるのを待って、ハンドルでコントローラの電源を入れればアップグレード完了です。



図 3-4 ファームウェアアップグレードの SCB 画面

サーボのアップグレード手順：

- 1) サーボアップグレードパッチを機器ローカルファイルにダウンロードします。
- 2) APP を開いてロボットに接続し、ロボットの電源が切れていること、緊急停止が解除されていることを確認します。
- 3) APP メイン画面の右上にある設定ボタンをクリックして、「システム設定」→「ファームウェアアップグレード」画面に入り、「ファイル選択」の空白部分をクリックして、サーボのアップグレードパッチファイル（パッチファイルの名前は変更せず、元の状態にしておくこと）を見つけてアップロードしてください。
- 4) アップグレード中は、コントローラの電源が自動的にオンして、APP 画面にアップグレードの進捗状況を示すウィンドウが表示され、アップグレードが完了するのを待ちます（数分程度）。そして、コントローラが自動再起動すると、アップグレード完了です。



図 3-5 ファームウェアサーボアップグレード画面

CAN カードのアップグレード手順：

- 1) CAN カードのアップグレードパッチを機器ローカルファイルにダウンロードします。
- 2) APP を開いてロボットに接続し、ロボットの電源が切れていること、緊急停止が解除されていることを確認します。
- 3) APP メイン画面の右上にある設定ボタンをクリックして、「システム設定」→「ファームウェアアップグレード」画面に入り、「ファイル選択」の空白部分をクリックして、CAN カードのアップグレードパッチファイル（パッチファイルの名前は変更せず、元の状態にしておくこと）を見つけてアップロードしてください。
- 4) アップグレード中は、コントローラの電源が自動的にオンして、APP 画面にアップグレードの進捗状況を示すウィンドウが表示され、アップグレードが完了するのを待ちます（数分程度）。そして、コントローラが自動再起動すると、アップグレード完了です。



図 3-6 ファームウェア CAN カードアップグレード画面

3.1.1.5 コントローラのアップグレード

ユーザーは、コントローラアップグレード画面で、ロボットのコントローラバージョンをアップグレードすることができます。図 3-7 に示した通り

正しいバージョンアップの手順は：コントローラのアップグレード→APP のアップグレード→SCB アップグレード→サーボのアップグレード→CAN カードのアップグレードとなりますが、その内数項目のみを対象としても、上記の手順に従ってください。

- 1) コントローラアップグレードパッチファイルを機器ローカルファイルにダウンロードします。
- 2) APP を開いてロボットに接続し、ロボットの電源がオフの状態にしておきます。
- 3) APP メイン画面の右上にある設定ボタンをクリックして、「システム設定」→「コントローラアップグレード」画面に入り、「ファイル選択」の空白部分をクリックして、コントローラアップグレードパッチファイル（パッチファイルの名前は変更せず、元の状態にしておくこと）を見つけてアップロードしてください。
- 4) コントローラが自動再起動すると、アップグレード完了です。



図 3-7 コントローラアップグレード画面

3.1.1.6 システム設定ファイル管理

システムバックアップには、設定ファイルのインポートとエクスポート及びプログラムの指定時間自動バックアップの二機能が含まれます。図 3-8 に示した通り。詳しい機能は以下の通り：

（1）設定ファイルのインポートとエクスポート

目的：設定ファイルとは、ユーザーがロボットを使用する際に修正したユーザー設定パラメーターのことで、中にはコントローラ設定パラメーター、IO 名前設定、ダイナミック IO 設定などが含まれます。ただし、全ての設定ファイルには、システム変数設定が含まれません。

エクスポート方式：エクスポートする必要がある構成ファイルを選択し、「ファイルのエクスポート」をクリックして、設定ファイルの保存先を選択し、「確認」をクリックするとエクスポートが完了します。

インポート方式：[ファイルのインポート]をクリックし、構成ファイルの保存パスを選択し、インポートする必要がある構成ファイルを選択して、[確認]をクリックします。インポート完了後に、コントローラを再起動します。注：異なるバージョンの設定ファイルを互いに読入すると、コントローラの異常起動になる可能性があるため、設定ファイルをインポートする際には、コントローラのバージョンを統一させるように注意します。

（2）指定時間自動プログラムバックアップ

目的：指定時間自動バックアップをオンにすると、プログラムが変更されると、APP は設定された時間に応じて計時を開始し、指定時間になると自動的に現在プログラムのバックアップ作業を行います。プログラム名は+“ ”という形式で、保存時のシステム時刻に基づいて命名されます。



図 3-8 システムバックアップ画面

3.1.1.7 権限暗証番号の変更

管理員は、ユーザー管理設定にて個々の操作権限暗証番号を図 3-9 に示した通りに変更できます。（デフォルト暗証番号：管理員：jakazuadmin 技術員：0000 操作員：0）。

注：この暗証番号は初期出荷暗証番号なので、初めての使用時に適時に変更してください。

設定画面に入り、「システム設定」を選択し、「ユーザー管理」をクリックし、変更したいユーザーレベルを選択して「変更」をクリックします。そして、画面の指示に従ってテキストボックスをクリックして入力し、「確認」をクリックすると変更が完了します。



図 3-9 暗証番号変更画面

3.1.2 操作設定

3.1.2.1 ツール座標系の設定

ツール座標系は、ツールセンターポイント（Tool Center Point、TCP）を原点に構築された座標系である。ロボット端末位置とは TCP が現在ユーザー座標系にあるデカルト座標のことであり、ロボット端末姿勢とはツール座標系が現在ユーザー座標系にある姿勢のことであり、RPY 方式で表します。JAKA Zu では、「TCP」がツール座標系を表します。。ロボットのデフォルトツール座標系は「フランジ座標系」となります。フランジ座標はロボット端末フランジのセンターを座標系原点とし、フランジ端

より外側向きは座標系Z軸正方向となり、フランジセンターとTIOとの接続ラインの方向が-Yとなり、右ねじの法則に従って+Xが決められます。

フランジ座標系のパラメーターは変更不可ですが、ロボット運動精度を高く求める作業では、TCPをチャック爪の末端センターや、吸盤のセンター位置などロボット端末アクチュエーターの末端部に設定するのが一般的です。従い、このデフォルトなツール座標系以外にも JAKA Zu ロボットは、パラメーター編集可能な TCP を十つ用意してあります。ユーザーは、自らのニーズに応じて「手動入力」、「四点設定」、「六点設定」などによる TCP パラメーター編集ができます。これより、上記三つの設定方法を説明します。

(1) 手動入力（図 3-10 に示した通り）：ユーザーは自らのニーズに応じて、必要なツール座標系がそれ相応なフランジ座標系にある位置と姿勢オフセットを計算したうえで、これらデータを対応するデータフレームに入力して「確認」を押すと、TCP パラメーターの編集が完了します。



図 3-10 ツール座標系の入力設定画面

(2) 四点設定（図 3-11 に示した通り）：確定空間内一固定点、ユーザー制御ロボット以 4 个不同的姿勢使 TCP 端末点到达该点置后即可通过「四点設定」自動計算出期望的ツール座標系相对フランジ座標系的位置と姿勢オフセット。具体的な設定プロセスは次のとおりです。

1. ロボットの手の届くところに、鋭い円錐の頂点など、固定された基準点を見つけます。
2. 設定した位置ポイント 1 をクリックして、ロボットを手動で操作して、アクチュエータの端末が基準点の位置に到達するようにします。「確定」をクリックする。
3. 設定した位置ポイント 2 をクリックして、ロボットを手動で操作し、アクチュエータの端末を位置ポイント 1 とは異なる関節角度で基準点位置に到達させ、[確認]をクリックします。
4. 設定した位置ポイント 3 をクリックして、ロボットを手動で操作し、アクチュエータの端末を位置ポイント 1 と位置ポイント 2 とは異なる関節角度で基準点位置に到達させ、[確認]をクリックします。
5. 設定した位置ポイント 4 をクリックして、ロボットを手動で操作し、アクチュエータの端末を位置ポイント 1、位置ポイント 2、位置ポイント 3 とは異なる関節角度で基準点位置に到達させ、[確認]をクリックします。
6. [確認]をクリックして、目的のツール座標系パラメータを取得します。



図 3-11 工具座標系の 4 点設定インターフェース

(3) 六点設定（図 3-12 に示した通り）：「4 点設定」操作に基づき、2 つの追加位置点を指定することで TCP 座標軸方向を決定し、TCP 位置と姿勢のパラメータの自動計算を完了し、目的ツール座標系のパラメータを自動計算します。六点設定はよくロボット端末ツールの軸ラインとロボット端末フランジとが垂直にならずか平行にならない場合に利用されます。「六点設定」を用いれば、ツール座標系 Z 軸とロボット端末軸ラインを一致させることが可能です。

具体的な設定プロセスは次のとおりです。

1. ロボットの手の届くところに、鋭い円錐の頂点など、固定された基準点を見つけます。
2. 設定した位置ポイント 1 をクリックして、ロボットを手動で操作して、アクチュエータの末端が基準点の位置に到達させ、[確認]をクリックします。
3. 設定した位置ポイント 2 をクリックして、ロボットを手動で操作し、アクチュエータの末端を位置ポイント 1 とは異なる関節角度で基準点位置に到達させ、[確認]をクリックします。
4. 設定した位置ポイント 3 をクリックして、ロボットを手動で操作し、アクチュエータの末端を位置ポイント 1 と位置ポイント 2 とは異なる関節角度で基準点位置に到達させ、[確認]をクリックします。
5. 設定した位置ポイント 4 をクリックして、ロボットを手動で操作し、アクチュエータの末端を位置ポイント 1、位置ポイント 2、位置ポイント 3 とは異なる関節角度で基準点位置に到達させ、[確認]をクリックします。
6. 設定した位置ポイント 5 をクリックして、位置ポイント 4 の位置と姿勢がそのままにしながら、位置ポイント 4 から、目的のツール座標系 Z 軸の正方向に沿って一定距離を移動して位置ポイント 5 を取得し、[確認]をクリックします。
7. 設定した位置ポイント 6 をクリックして、位置ポイント 5 の位置と姿勢がそのままにしながら、位置ポイント 5 から、目的の X O Y 平面に一定距離を移動して位置ポイント 6 を取得し、[確認]をクリックします。
8. 「確認」をクリックすると、希望するツール座標系のパラメーターを取得できます。上記位置ポイント 4 と位置ポイント 5 との接線方向は、即ち希望するツール座標系 Z 軸の正方向となります。



図 3-12 ツール座標系の六点設定画面

3.1.2.2 負荷情報設定

負荷とは、ロボット端末にキャリアさせる全ての物体の質量とセントロイド位置のことを指します。正確な負荷情報を設定することによって、コントローラにロボットの実際動作状況を正しく計算できるようになります。負荷の設定が実際の状況と大きく異なる場合、ロボットの運動中にコントローラが衝突を誤って認識し、ロボットの運動が中断されることがあります。

負荷の設定方法は二つがあります：手動入力法と自動認識法。各々のモードにおいて、負荷を設定する手順は以下の通りとなります。

(1) 手動入力モード（図 3-13 に示した通り）：

ユーザーは、計算または測定に基づいて正確な荷重情報を取得し、対応する入力枠に負荷の質量とセントロイド位置を正しく入力し、「確認」をクリックして設定を完了します。

注意：セントロイド位置はロボット端末フランジのセンター部に準じたものであり、X、Y、Z もフランジ座標系の空間値を表します。通常、我々は正確度を確保するために、ユーザーには 3D 設計ソフトウェアを使用して計算を実行することを勧めます。



図 3-13 負荷の手動入力設定画面

(2) 自動認識モード（図 3-14 に示した通り）：

目的：固定位置からのロボット運動に基づいて、負荷の質量とセントロイド位置を自動的に識別し

て計算します。

設定手順：

1. ロボットの各関節が既にゼロ校正済みのこと；負荷が正しくセッティングされること、端末ツールの場合、取付姿勢と実質稼働と一致していること。
2. 「自動認識モード」をクリックして、2 パターンの負荷認識に入ります。：有負荷・無負荷。まず「はい」をクリックして有負荷識別を行います。有負荷識別が完了しないと無負荷識別に移ることができません。負荷識別時において、負荷とロボット本体との干渉を避けるように、ユーザーは「開始ポイント」を設定する必要があります。
3. 3 号軸の開始ポイントを設定する場合、3 号関節角度が $\leq -50^{\circ}$ で、4 号関節が 90° 、5 号関節が $\pm 180^{\circ}$ 、6 号関節が 0° となるようにすること；4 号軸の開始ポイントを設定する場合、4 号関節角度が $-30^{\circ} \sim 30^{\circ}$ の間で、5 号関節が $\pm 180^{\circ}$ 、6 関節が 0° となるようにすること；5 号軸の開始ポイントを設定する場合、2、3、4 号関節が何れも 90° で、5 号関節が $\leq 70^{\circ}$ 、6 号関節が 0° となるようにすること；6 号軸の開始ポイントを設定する場合、5 号関節が $\pm 180^{\circ}$ となるようにします。上記開始ポイントの必要条件が満足できなければ、APP に注意情報が表示されます。
4. 開始ポイントの設定を正しく完了した後、「テスト点実行」をクリックすると、当該関節が現在位置から $+190^{\circ}$ 回転しますが、ロボットの実行中に干渉物がないことを確認した後に、「開始ポイント設定」を長押しして初期位置に戻してから、「識別開始」をクリックします。識別が終わった後に、次の関節の識別に移ります。
5. 負荷識別が完了したら、負荷を取り除き、「いいえ」をクリックして無負荷識別に移ります。有負荷識別の動きが支障なく確認できたので、ロボットが初期位置に戻るまで「開始ポイント設定」を長押しして、「識別開始」をクリックします。4 つの関節の動きがすべて確認できたら、「識別結果の取得」をクリックすると、APP から荷重質量とセントロイドのパラメータが表示されます。それを確認すると、コントローラに更新させたり、手動入力モードで表示させたりすることができます。
6. 識別中に、「リセット」をクリックすると、今までの識別結果を全てクリアして、再識別ができるようになります。



図 3-14 負荷自動識別設定画面

3.1.2.3 ユーザー座標系設定

ロボットのデフォルトユーザー座標系は世界座標系となります。世界座標系は、ロボットのベース中心点を原点に、土台からロボット本体への垂直方向を+Zとし、土台中心点と重負荷ラインのインターフェースとの接続方向を+X 方向とし、右ネジ法則に基づいて+Y が決められますが、当該 TCP パラメーターは変更不可となります。

世界座標系以外にも、JAKA Zu ロボットは、パラメーター編集可能なユーザー座標系を別途十つ用意してあります。ユーザーは自らのニーズに応じて「手動入力」や「三点設定」などを用いて、ユーザー座標系パラメーターを編集することができます。これより、上記三つの設定方法を説明します。

(1) 手動入力（図 3-15 に示した通り）：ユーザーは自らのニーズに応じて、必要なユーザー座標系が世界座標系に対応する位置と姿勢オフセットを計算し、これらデータを対応するデータフレームに記録して確定をクリックすることによって、ユーザー座標系パラメーターを編集することができます。



図 3-15 ユーザー座標系入力設定画面

(2) 三点設定（図 3-16 に示した通り）：

目的：ユーザーが設定した 3 つの位置ポイントに基づき、自動的に希望するユーザー座標系の X、Y、Z 方向上のパラメーターを計算します。三点設定によって生成した TCP の X、Y、Z 方向を世界座標系と一致させます。

三点の定義：

位置ポイント 1：期望ユーザー座標系の元点。

位置ポイント 2：希望するユーザー座標系 X 軸の正方向にある任意ポイント。

位置ポイント 3：希望するユーザー座標系 XY 平面第一象限にある任意ポイント。



図 3-16 ユーザー座標系の三点設定画面

3.1.2.4 据付姿勢設定

助ちゃんロボットは任意角度での据付に対応可能です。助ちゃんロボットの据付が完了できたら、実行中の安全性を考慮して助ちゃんの据付姿勢データをソフトに入力する必要があります。機能バーの「設定」をクリックし、「据付設定」を選択して、実際の助ちゃんロボットの据付姿勢に応じてソフトウェア内の模擬ロボット据付姿勢を調整する目的で、右上より反面、側面か前面（デフォルトは前面）を選択します。そして、方向性を示す三角ボタンをクリックして据付角度を選択して、ソフトウェア内の模擬ロボット据付姿勢が実際の状況と一致させるようにしたうえで、「確認」をクリックして有効にします。図 3-17 に示した通り。



図 3-17 据付設定画面

3.1.2.5 故障情報の取得仕方

ロボットが故障した場合、コントローラは故障関連情報をその時点のシステム時刻を名称に自動的に圧縮ファイルを生成して、故障診断画面に表示します。図 3-18 に示した通り。

ただし、ロボットが故障しなくても、ユーザーは故障診断画面にて、「故障記録」ボタンをクリックできますので、その場合、APP には「故障診断開始」、ログボタンの隣に「故障診断中」というダイアログが現れますが、約 30 秒を待ってから、故障診断が自動的に完了します。④「故障診断」ボタンまたは停止ボタンをクリックすると、事前に故障診断を停止することができます。⑤



図 3-18 故障診断画面

3.1.3 安全設定

3.1.3.1 関節リミットポジション設定

関節リミット画面にて、ユーザーは各々の関節のソフトリミット角度、関節スピード制限、誤差警報範囲を設定することができます。図 3-19 に示した通り。

注：誤差警報範囲とは、機械アームの運動偏差誤差値が警報範囲を超えた場合、ロボットが警報して喚起する機能です。

注意：関節プラスリミット、関節マイナスリミット、関節スピード制限デフォルト値がいわゆる最大範囲であり、ユーザーはデフォルトの範囲内のみ修正できます。（1.5 バージョン一、五、六関節リミットを $\pm 360^\circ$ に変更；APP に表示されるデフォルト値は相変わらず $\pm 270^\circ$ ）

関節の名称	関節のプラスリミット	関節のマイナスリミット	関節スピードリミット	誤差アラーム範囲	リセット
関節一	360.000 °	-360.000 °	120.000 °/s	80 %	⌂
関節二	265.000 °	-85.000 °	120.000 °/s	80 %	⌂
関節三	175.000 °	-175.000 °	120.000 °/s	80 %	⌂
関節四	265.000 °	-85.000 °	150.000 °/s	80 %	⌂
関節五	360.000 °	-360.000 °	150.000 °/s	80 %	⌂
関節六	360.000 °	-360.000 °	150.000 °/s	80 %	⌂

図 3-19 関節リミット画面

3.1.3.2 ロボット姿勢設定

ロボット姿勢画面には三つ姿勢があります：出荷姿勢、開放姿勢、初期姿勢。図 3-20 に示した通り。

出荷姿勢：ロボットの箱詰め姿勢です。

開放姿勢：ゼロ位置と姿勢で、関節のゼロ復帰に使用します。

初期姿勢：ユーザー編集用の安全姿勢で、ユーザーはある安全姿勢を初期姿勢に設定したり、ロボットハンドルを用いて初期姿勢に操縦したりすることができます。ロボットが初期姿勢になった場合、機能 IO の「安全位置」がトリガーできます。

変更ボタンをクリックして姿勢位置を変更します。✎

「目標ポイントへ移動」を長押しして、ロボットを目標位置へと移動します。



図 3-20 ロボット姿勢設定画面

3.1.3.3 保護システム設定

JAKA ロボットは外部視覚を持つことで安全保護状態を実現しています。ユーザーは実際の作業条件に応じて減速エリアと協働エリアを定義することで、人や外部の物体が対応するエリアに入ると、ロボットは認識して適宜に減速または停止状態に入り、不用意な侵入による安全事故を防ぐことができます。

操作手順：

①まず APP にて拡張 I/O 設定（図 3-21.1-----3-21.4 に示した通り）を行います。

以下の手順で APP に拡張 I/O を設定します。拡張 I/O を設定して起動すると、コントローラは視覚保護との通信を確立し、視覚保護装置のモニタリング情報を得ることができます。拡張 I/O 的設定手順は以下の通りですが、IP アドレスとポート号は完全に図の通りに従っても良い。

拡張 I/O の追加が完了したら、DI0 を減少モードにし、DI1 を保護的停止モードに設定します。これで拡張 I/O 機能の設定は完了です。視覚保護機能がオンになっていることを確認した後、拡張 I/O を動作状態に切り替えて、視覚保護装置とコントローラ間のリアルタイム通信を確立することができます。

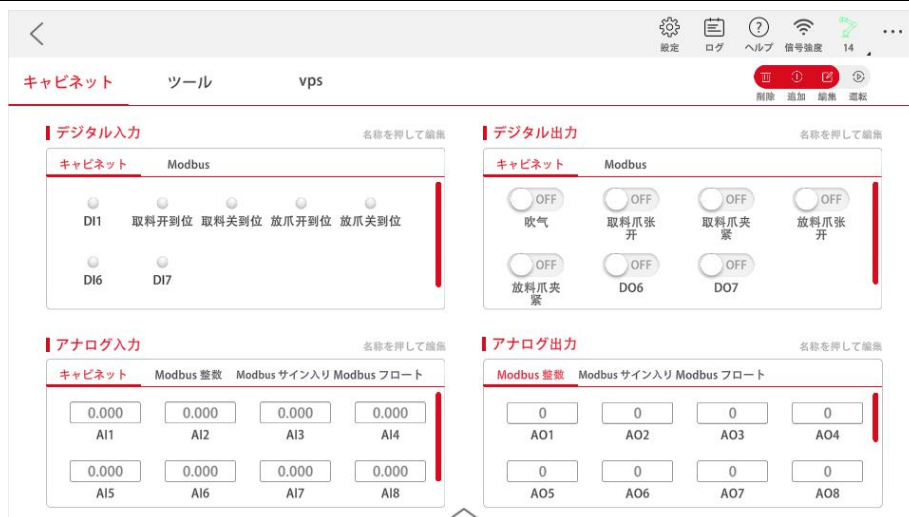


図 3-21.1 視覚保護 I/O 設定画面



図 3-21.2 視覚保護 I/O 設定画面



図 3-21.3 視覚保護 I/O 設定画面



図 3-21.4 視覚保護 I/O 設定画面

②APPにて保護エリアのカスタマイズ設定（図 3-22 に示した通り）を行います。

APP 端末より保護エリアのカスタマイズが可能になり、エリア設定も複数の形状に対応するため、自由に選択できます。

図 3-22 に示した通り、APPの視覚保護モジュールを開き、接続設定ボタンをクリックします。視覚保護装置のIPと設定したい保護エリアの形状を記入することで、保護装置に接続できます。本品は長方形、円形及びカスタムカーブという三種類の保護エリア形状に対応できます。

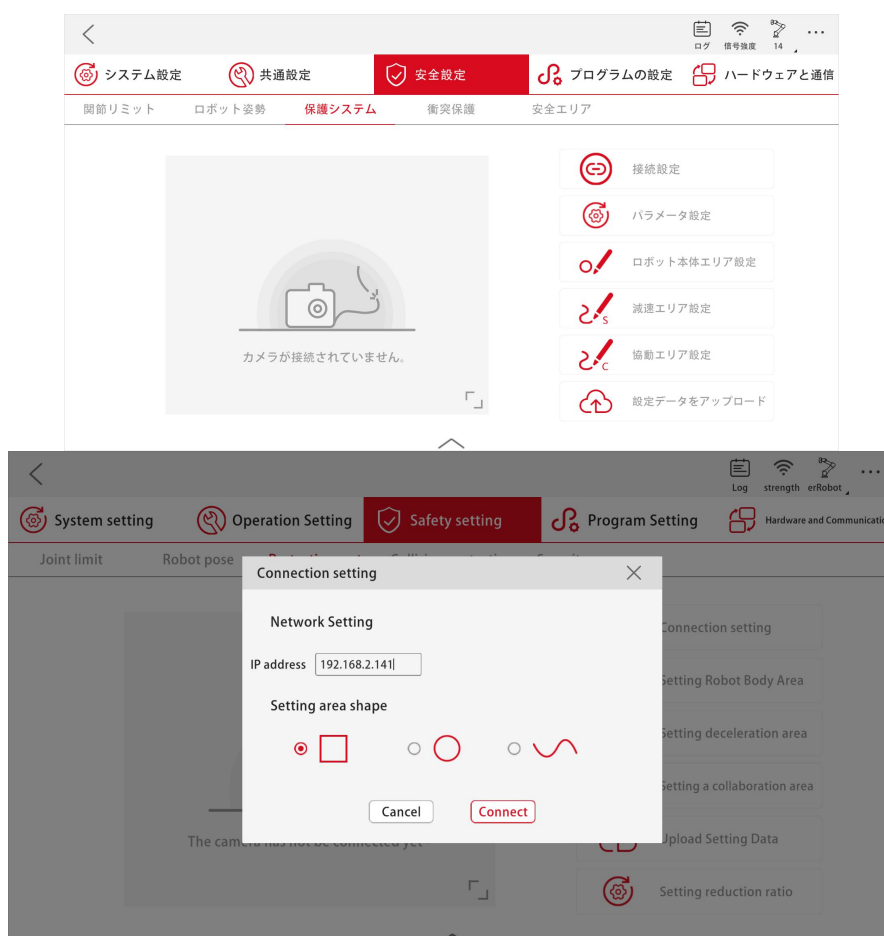


図 3-22 視覚保護接続設定画面

当該モジュールに設定する必要があるエリアは以下三つが含まれます：ロボット本体エリア、減速エリア、協働エリア。
図 3-23.1---3-23.3 に示した通り。三つエリアの働きはそれぞれ：

ロボット本体エリア

名のとおり、ロボット本体が占めるエリアのことです。実際のプログラム実行時にはロボットが動いているので、ロボットを侵入物として検出しないためには、ロボットの活動エリアをアルゴリズムから除外する必要があります。当該モジュールでは、ロボット本体のエリアは円形を描く方法でしか決められません。設定方法：ロボットが実際に実行するジョブプログラムを実行し、視覚保護設定画面にてロボット本体エリアを設定し、ロボットが常時設定された本体エリアの円形内にいることを確認します。

減速エリア

当該エリアは、第一級保護エリアです。物体が当該エリアに侵入してくると、視覚保護装置がコントローラに情報を発信し、コントローラはロボットを縮小モードに制御します。当該エリアの設定はユーザーが設定時に選択した形状に基づいて実行します。下図のように、設定時には長方形の設定モードを選んだので、カスタマイズ減速エリアの形状も長方形になります。エリアを選んだ後、設定データアップロードをクリックして、設定情報を CPU に送るとともにローカルに保存します。アップロードしてエリア設定を一度に有効すると、次回に視覚保護システムの起動にもそのまま反映されます。エリア設定を変更したい場合は、再度設定し直してアップロードすればよい。

協働エリア

当該エリアは、第二級保護エリアです。物体が当該エリアに侵入してくると、視覚保護装置がコントローラに情報を発信し、コントローラはロボットを保護的停止モードに制御します。当該エリアの設定はユーザーが設定時に選択した形状に基づいて実行します。下図のように、設定時には長方形の設定モードを選んだので、カスタマイズ減速エリアの形状も長方形になります。エリアを選んだ後、設定データアップロードをクリックして、設定情報を CPU に送るとともにローカルに保存します。アップロードしてエリア設定を一度に有効すると、次回に視覚保護システムの起動にもそのまま反映されます。エリア設定を変更したい場合は、再度設定し直してアップロードすればよい。



図 3-23.1 視覚保護カスタマイズエリア設定画面



図 3-23.2 視覚保護カスタマイズエリア設定画面



図 3-23.3 視覚保護カスタマイズエリア設定画面

視覚保護機能のオンオフは、APP での拡張 I/O が稼働モードに設定されている場合のみ、ロボットに視覚保護機能を発効させることができます。視覚保護機能を無効にしたい場合は、拡張 I/O を編集可能状態にするか、ネットワークケーブルを切断することで、視覚保護プログラムの実行を停止することができます。

注：

- a カスタマイズエリア設定時に干渉がないことを確認します。
- b このアルゴリズムでは、設定時のリアル画像を参照画像として使用して視覚安全保護を行います。従い、カスタマイズエリアを設定する際には、当該エリアに他の物体が干渉していないこと、即ち現時点に物体侵入がない状態にしておく必要があります。
- c カスタマイズエリア設定のロジックが正しいこと。
- d 協働エリアの保護水準が減速エリアより高いため、カスタマイズエリアの設定を行う際は、協働エリアが減速エリアの内側にあるようにしておく必要があります。カスタマイズ曲線を使ってエリア設定を行う場合、描かれた曲線が極力閉鎖したエリアになるようにする必要があります。

3.1.3.4 衝突レベル設定

衝突保護設定画面には、二つの設定方式があります：クイック設定とカスタマイズ。図 3-24 に示した通り。

(1) クイック設定：ユーザーは、クイック設定にて素早くロボットの衝突保護感度を設定することができます。

(2) カスタマイズ：ユーザーはカスタマイズにて、自らのニーズに応じて衝突保護感度を設定することもできます。その中で、力の限界とは、衝突力の限界レベルを指します。

注：力制限レベル 1-5、制限条件はやや緩い、レベル 0 は力制限をなくすこと。動量、TCP スピード、パワーの制限値が小さいほど、ロボットの運動速度は遅くなります。リミット値はユーザーの実情に合わせて設定する必要がありますが、特別な必要性がない場合は、より大きなリミット値を設定することを勧めます。

動量制限、TCP スピード制限、パワーの制限値が小さいほど、ロボットの運動速度は遅くなりますので、ユーザーは自らのニーズに基づいて適宜な数値に設定してください。



図 3-24 衝突保護設定画面

3.1.3.5 安全エリア設定

ロボットが運動中に他の物体との衝突を避ける目的により、ユーザーはロボット端末が常時設定済み安全エリアに停止させるための安全エリアを設定することができます。

安全エリアは、二つの有効方式があります：起動による有効、実行による有効。起動による有効とは、ロボット起動時に、安全エリアを有効にすることを意味します、ロボットが安全エリアに到達した途端に運動が停止します；実行による有効とは、ロボットがプログラムを実行する場合に安全エリアを有効にし、ドラッグ、手動操作モードでは、安全エリアが有効にしません。図 3-25 に示した通り。

ユーザーは、のべ 6 つの安全エリアに対して、以下の設定方法に従って設定できます。：

1. 設定したい安全水平面オプション右側の矢印をクリックして、編集シートを開きます。
2. 水平面名称を入力します。
3. 水平面ポイント 1 の右側の歯車アイコンをクリックし、手動操作画面に入ります。そして、ロボットを希望する安全エリアに移動させてから「確定」をクリックします。
4. 水平面ポイント 2 の右側の歯車アイコンをクリックし、手動操作画面に入ります。そして、ロボットを希望する安全エリアに移動させてから「確定」をクリックします。
5. 水平面ポイント 3 の右側の歯車アイコンをクリックし、手動操作画面に入ります。そして、ロボットを希望する安全エリアに移動させてから「確定」をクリックします。コントローラは三つのポイントに基づき、安全エリアの空間位置を計算したうえで、左にある三次元モデルにて表示します。
6. 安全面ポイント設定は、安全エリアの安全面を確認するためのものです。安全面ポイントの右側の歯車アイコン

をクリックし、手動操作画面に入ります。そして、ロボットを希望する安全エリアに移動させてから「確定」をクリックします。

7. 「有効にするか」の欄で、当該安全エリアを有効にします。

8. 安全距離とは、ロボット端末と安全エリア間のことで、単位は mm とします。安全距離がユーザー設定値と同一かそれ以下となった場合、安全エリアがトリガーされ、ロボット運動が停止します。



図 3-25 安全エリア画面

3.1.3.6 安全単独 IO 設定

注：この機能は 2.1 バージョンの制御盤のみに対応可能で、他のバージョンの制御盤には対応しません。

安全設定では、コントローラパネルの物理的な I/O を安全信号との紐付けとして、制御盤 I/O をロボットの安全動作を制御したり、ロボットの安全状態を監視したりするための単独の安全 I/O 信号に設定することができます。この単独安全 I/O 信号は二回路タイプのもので、安全制御と安全状態は図 3-26 に示した通り：



図 3-26 安全信号説明画面

単独安全 I/O を設定するには、まずロボットの電源が切れていることを確認し、設定をクリックして安全設定に入り、単独安全 I/O をクリックして、必要な I/O を選択し、ドロップダウンライアングルをクリックして、希望する機能を選択し、確認をクリックして有効にします。図 3-27 に示した通り。



図 3-27 単独安全 IO 設定画面

3.1.4 プログラム設定

3.1.4.1 起動デフォルトプログラム管理

プログラム設定において、ユーザーは特定のプログラムをデフォルトプログラムに指定することができます。プログラム設定画面のオプションより、ロボットが起動時に自動的にデフォルトプログラムをアップロードさせます。これにより、ロボットの電源を入れるたびに APP を介してロボットにログオンする、電源を入れる、起動させる、実行するプログラムを選択するという一連の作業を減らすことができるというメリットがあります。ロボットの電源を入れてから、直接ハンドルからロボットの電源を入れたり、起動させたりすることができ、ハンドルのプログラムスタートボタンや起動 I/O での「起動プログラム」を使って、ロボットのデフォルトプログラムをそのまま実行することができます。図 3-28 に示した通り。



図 3-28 デフォルトプログラム設定画面

3.1.4.2 軌跡記録機能

「軌跡記録」設定画面では、ユーザーはドラッグ、手動操作などにより、コントローラにその間に起きたロボットの移動軌跡を記録させ、軌跡ファイルとして生成することができます。プログラミング時には、軌跡記録コマンドを用いてこれら軌跡ファイルを読み出し、ロボットにティーチングされた軌跡をプログラムに再現させることができます。（注：軌跡再現

機能は記録のパス情報のみを記録し、軌跡のスピード情報は記録しません。) 図 3-29 に示した通り

運動軌跡の記録方法：

1. 設定ボタンをクリックして、軌跡運動編集画面に移ります。 この画面では、制御装置が軌道をどの程度の速度と加速度でサンプリングするかを設定したり、位置や姿勢のサンプリング精度を設定することができます。軌跡の運動距離が短すぎる場合、位置と姿勢のサンプリング精度を 0.1 に引き上げることを勧めます。

2. 追加ボタンをクリックして画面からの「軌跡記録を始めるか」ダイアログに「はい」をクリックします。同時に、APP のトップ部状態バーには「軌跡記録中」ボタンが表示されます。

3. ドラッグか手動操作で、ロボットに希望する動作軌跡を取らせます。

4. 動作ティーチング後、「軌跡記録中」ボタンをクリックすると、運動軌跡の記録が停止するとともに、運動軌跡画面にて軌跡ファイルが生成されます。

5. 編集ボタンをクリックすると、軌跡ファイルの名前が変更できます。

6. プログラミング制御画面にて、軌跡再現プログラムを読み出して実行させることができます。



図 3-29 軌跡記録の設定画面

3.1.4.3 システム変数の新規作成仕方

システム変数は数値のみで、コントローラに独立して格納されており、プログラムの開始・終了、ロボットの電源の有無、電子制御盤の電源の有無などによって変数の値が変化したり、リセットされたりすることはありません。同時に、システム変数は任意単独なプログラムに読み出されたり、修正されることができます。図 3-30 に示した通り

システム変数の追加方法：プログラム設定-システム変数にて、追加をクリックして、システム変数編集画面に移ります。 編集画面に変数名と初期値を入力して確定をクリックすると、システム変数の追加が完了します。

注：システム変数は最高 100 個まで格納できます。



図 3-30 システム変数の設定画面

3.1.5 ハードウェアと通信

3.1.5.1 Modbus 通信設定

JAKA Zu シリーズロボットは Modbus TCP/IP と Modbus RTU 通信方式に対応しており、この設定はロボットのための Modbus Server 側の設定画面となります。詳細な操作手順は以下の通り：

Modbus TCP/IP

APP から設定→ハードウェアと通信→Modbus パラメーター設定に入り、Modbus TCP/IP モードを選び、パラメーターを修正してから確定をクリックし、設定成功を確認してから、制御盤を再起動すると接続できます。図 3-31 に示した通り

ポート番号：Client ポート番号と一致



図 3-31 Modbus パラメーター画面

Client 側が接続を確立した後は、Modbus Address テーブルのレジスタアドレスや機能コードに基づき、ロボットの状態を読み取ったり、ロボットの I/O 信号を制御するためのプログラムを編集する必要があります。APP の I/O モニタリング→コントローラ→Modbus で確認できます。図 3-32 に示した通り。Modbus Address テーブルの詳細は付録に参照します。

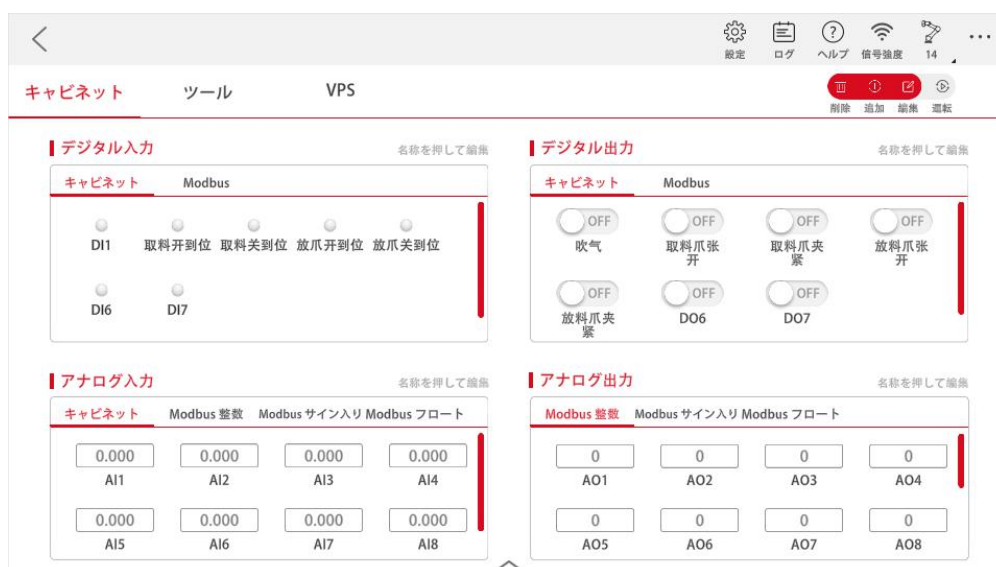


図 3-32 IO パネル画面 Modbus 画面

Modbus RTU

設定画面を開き、ロボットを Modbus Slave のパラメーターとして設定します。PLC 側設定画面は、ロボットを Slave パラメーターとして設定する必要があります。図 3-33 に示した通り。



図 3-33 Modbus パラメーター画面

Client 側が接続を確立した後は、Modbus Address テーブルのレジスタアドレスや機能コードに基づき、ロボットの状態を読み取ったり、ロボットの I/O 信号を制御するためのプログラムを編集する必要があります。APP の I/O 画面での Modbus から確認できます。図 3-32 に示した通り。Modbus Address テーブルの詳細は付録に参照します。

3.1.5.2 端末（力制御）センサー

1. JAKA Zu se システム構築

図 3-34、3-35 に示した通り、ユーザーが簡単なハードウェア接続とソフト設定を行うことで、手軽に JAKA Zu se システム構築を実現できます。詳しい手順は以下の通り：

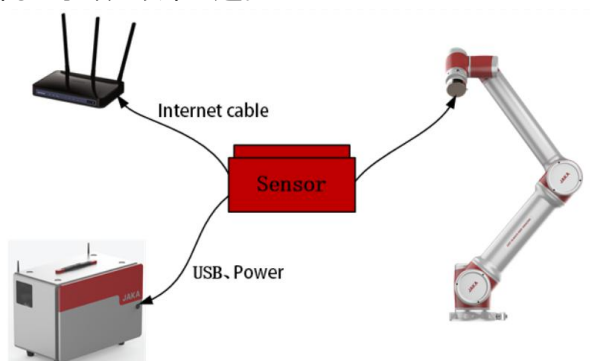


図 3-34 JAKA Zu se 構築イメージ図



図 3-35 力制御センサー設定画面

a. ユーザーは、実際のニーズに応じて力制御センサーと末端フランジ間の接続品を設計することにより、接続品がツール座標系の設計を介してセンサーの X+ 方向とロボットフランジの X+ 方向と一致させることができます；

b. 力制御センサーは、ロボットコントローラより、或いはユーザーによる外付けの 24V の DC 形式より給電することができます。センサーはその通信方式により、制御盤の USB ポートを介したシリアル通信、或いはネットワークポートを介した TCP 通信をサポートします；

c. センサー I や III タイプの場合、APP にて IP とポート番号を事前設定する必要がありますが、センサー II の場合、それが一切不要となります。関係センサー型番の詳細は付録に参照してください。

2. JAKA Zu se の取扱方法

a. 自動認識モード

力制御制御関連の機能を使用する際には、まず正確な負荷パラメータを設定する必要があるため、[負荷設定] から手動入力を選ぶか、下記手順となる自動認識機能を使用することもできます。

【開始位置設定】より手動画面に入り、ロボットを適当な位置に運動させてから終了を確認します。

【終了位置設定】より手動画面に入り、4、5、6 号関節のみ初期位置の $\pm 90^\circ$ 以内に（ロボットの運動が干渉しない前提で、運動範囲が大きいほど、認識結果もより精確になる）運動させて終了を確認します；

【テスト点実行】、【開始位置設定】を長押しして初期位置に戻り、【テスト点実行】を長押しして軌跡に干渉しないことを確認します；

[識別開始]、[開始位置設定] を長押しして初期位置に戻り、[識別開始] をクリックして数秒待つと、APP に識別結果が表示され、間違いなく確認します；問題がある場合、（再度）終了して、再度センサー設定画面にて認識し直します。

b. 安全保護機能

図 3-36 に示した通り、在【センサーリミット】にて保安上力制御制限値を設定しておけば、ロボット運動中にセンサー末端に掛かる負荷が設定範囲を超えた場合、予想外の危険や経済的損失を防ぐようにロボットが直ちに運動を停止します。



図 3-36 JAKA Zu se 安全保護機能の設定

c. 牽引ティーチング機能

図 3-36 に示した通り、センサー種類を選んで IP、ポート番号、フィルタリングと負荷などを設定し終わった後に、【確定】して【実行】させると、コントローラがセンサーからデータを受信し始めます；

図 3-37 に示した通り、【手動操作】に入り、まず【設定】画面にて F_x 、 F_y 、 F_z は各々 X、Y、Z という三方向の変位を、 M_x 、 M_y 、 M_z は各々 X、Y、Z という三方向の回転を表します；【方向】前にある ☐ をチェックさせることで、その一つや複数方向を起動させる、即ちドラッグ起用後に、起動された方向でロボットをドラッグすることができます；【減衰力】の設定値が低いとユーザーのドラッグ必要値もその分低くなりますが、【減衰力】設定値が無限に低いということではなく、 F が 10N 以上、 M が 0.2Nm 以上にすることを勧めます。また設定値は 0 にしてはなりません；【跳ね返り】は、ロボットをドラッグ前の初期位置に戻すことが可能ですが、設定値が高ければ、必要とするドラッグ力もその分高くなります；

力制御座標系、【ツール座標系】にチェックを入れると、設定されたツール座標系にてロボットをドラッグします；【世界座標系】にチェックを入れると、世界座標系にてロボットをドラッグします；

初期化するか否か、コントローラ実行後に初めてドラッグモードに入る場合は、【初期化】にチェックを入れ、センサーのオフセットと負荷を補正する必要があります。また、補正の精度を影響しかねないので、ドラッグの実行中に、ロボット端末に外部力との接触を避けてします。

パラメーター設定を終了したら【確定】を、そして【終了】をクリックして【ドラッグ】モードに入ります。ただし、その前には、センサー補正値の偏差や、危険を防ぐように、ロボット端末に外力が受けないようにします。

センサーに温度ドリフト現象が存在するため、ドラッグモードにてロボット位置に変位が現れたら、一旦【終了】して再度【ドラッグ】に入り、センサー補正をやり直してください；牽引ティーチング機能を利用しない場合は、速やかに【終了】してください。



図 3-37 JAKA Zu se 牽引ティーチング機能の設定

d. 一定力モード

一定力モードは、ロボット端末と外部環境の間の接触力を設定された一定力値範囲内にすることができます。図 3-38 に示した通り、一定力モードには【一定カスムージング制御有効】、【力制御座標系設定】、【一定カスムージングパラメーター設定】と【一定カスムージング制御無効】が含まれます。【一定カスムージングパラメーター設定】画面にて牽引ティーチングパラメーターと一致するパラメータを設定します。ユーザーは、希望する接触力値に基づいた【一定力】値設定ができます。一方、【減衰力】の設定は外部環境の耐性に依存していますが、一般的には耐性が大きいほど、必要な【減衰力】値もその分高くなります。センサーに温度ドリフト現象が存在するため、ロボットを一定時間に実行した後は、ロボット端末に外力との接触が起きないように、【初期化】にてセンサーを補正し直すことを勧めます。



ie 一定力モードのプログラミング

e. スピードモード

スピードモード、ロボット端末の負荷力が制御力の設定値を超えた場合、ロボットはセンサー検出値が制御力設定値以下になるまで減衰します。図 3-39 に示した通り、スピードモードには、【スピードスムージング制御有効】、【スピードスムージングパラメーター設定】、【スムージング制御力】と【スピードスムージング制御を閉じる】が含まれます。【スピードスムージングパラメーター設定】画面で設定スピード段差を、【スムージング制御力】画面で制御力の範囲を設定します。

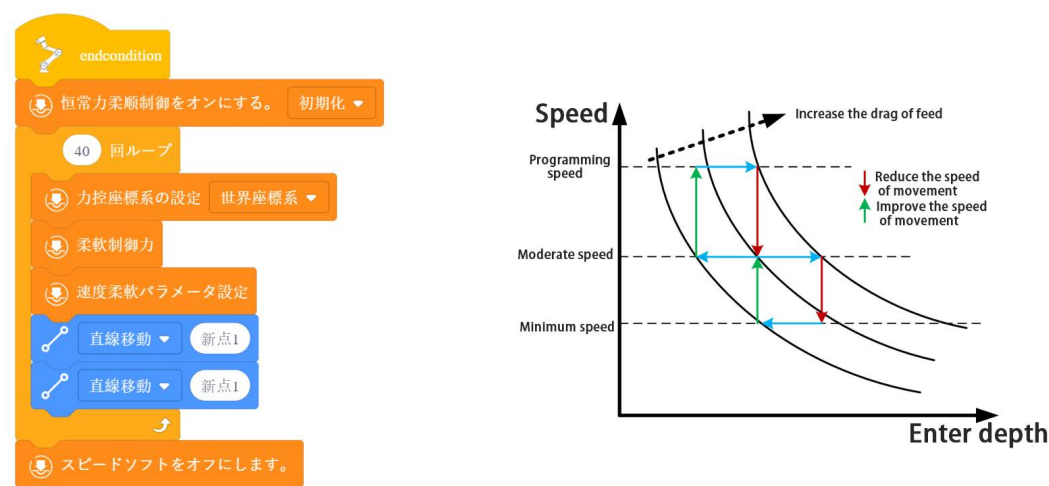




図 3-39 JAKA Zu se スピードモードのプログラミング

f. 運動終了条件

運動終了条件設定画面にて、モニタリングする方向を選択し、上限値や下限値を設定します；外部接触値が下限値や上限値を超えた場合、運動終了条件がトリガーします。図 3-40 に示した通り、【運動終了条件】モニタリングの直近の次なる運動コマンドによって運動終了条件がトリガーされた場合、ロボットは直ちに現在位置から次なる運動コマンドの終了位置まで運動します。必ず【一定カスミングパラメーター設定】にて全パラメーターを 0 に設定したうえ、任意方向にもチェックを入れないことに注意してください。



図 3-40 JAKA Zu se 運動終了のプログラミング

g. 外力のリアルタイム表示

手動画面にて【表示】を選ぶと、接触外力値をリアルタイムに観察することができます。ただし、手動モードでは、力制御制御【ドラッグ】モードでのみ接触外力値が表示されます。

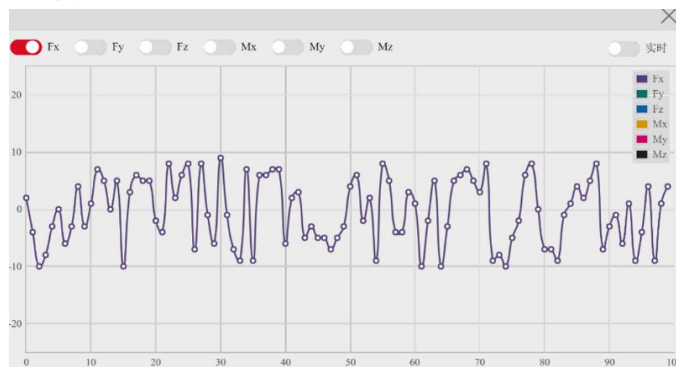


図 3-41 APP の力表示画面

3. JAKA Zu se 取扱注意事項

a. 内蔵型力制御センサーは精密機器ですので、必ず取扱マニュアルに記載されている仕様の範囲内で使用してください。特に定格荷重以上の負荷で使用すると、製品の故障の原因となりますので、力制御センサーの全方向が負荷範囲内にあることを確認してください。

b. ロボットセンサー端末の負荷設定が正しいことを確認してください；

c. センサーの X+ 方向とロボットフランジの X+ 方向が一致していること、またツール座標系を設定することによって方向の一致性を確保してください；

d. 力制御制御に入る前に、ロボットの端末と外部環境との間に接触力が存在しないようにしてください。

3.1.5.3 端末 IO

注：本機能は V3 バージョンの端末 TIO のロボットバージョンのみに適用します。

ロボット端末（ツール側）に配置されている小型 IO インターフェース（TIO）は、デジタル入力とアナログ入力を 2 経路ずつ対応します。その内、デジタル入力の 2 経路は高速 RS485 チャンネルとして多重化でき、アナログ入力の 2 経路は低速 RS485 チャンネルとして多重化できます。同時に、設定可能な電圧出力（12V/24V/0V）にも対応し、外部拡張装置への給電として利用できます。そして、APP 設定→ハードウェアと通信→端末 IO にて設定できます。

①電圧出力設定

ツール側 TIO が持つ端末電源出力機能を利用すると、ユーザーは電圧出力するか否かを選び、起動電圧の出力値を 12V か 24V に設定することができます。【設定】->【ハードウェアと通信】->【電圧出力】画面にて、起動電圧の出力及び出力値の設定を、下図 3-42 に示した通りに行ってください。



図 3-42 TIO 電源出力機能の設定

②RS485

RS485 設定

2 経路の RS485 チャンネルは、対応するピンが RS485 チャンネルとして多重化された後にのみ設定できます。RS485 チャンネル 1 を例とした場合、まず DO ピンを図の通りに、RS485 チャンネル 1 に多重化します。設定完成后、RS485 チャンネルの設定オプションボタンが図の通りに表示されます。【RS485 設定】をクリックすると、RS485 チャンネルの設定ダイアログ画面が表示され、それには図 3-43、3-44、3-45 に示した通り、通信パラメーター及び機能モード設定が含まれます。

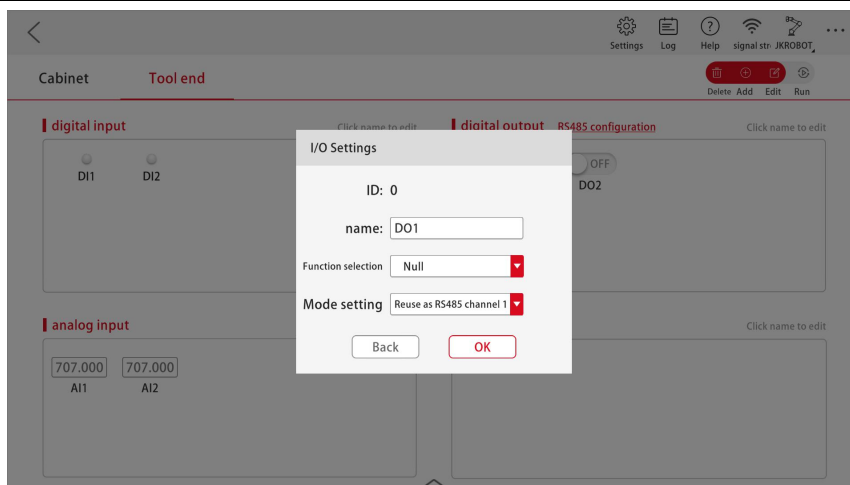


図 3-43 デジタル出力を RS485 チャンネル 1 チャンネルに多重化する

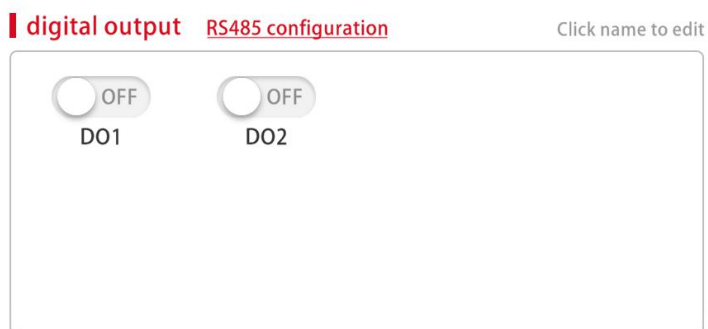


図 3-44 RS485 設定オプション

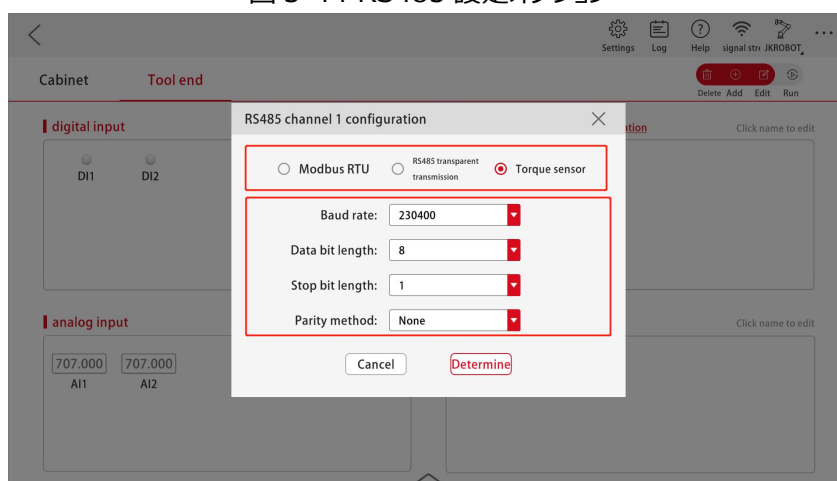


図 3-45 RS485 チャンネル設定画面

RS485 チャンネル設定：

RS485 チャンネルを利用する場合モード設定が必要ですが、現在三つのモードが設定できます：通過性トランスポートモード、Modbus RTU モード及びトルクセンサーモード、図の上部にある赤枠に示した通り：

- 通過性トランスポートモードは今後のバージョンアップ専用として、現在如何なる装置にも対応していません。
- Modbus RTU は主に外部チャック爪などの装置に利用されます；
- トルクセンサーモードは、主に指定型番のトルクセンサーの接続用に利用されます。

注意：現在使用可能な 2 経路 RS485 チャンネルについて、トルクセンサーは RS485 チャンネル 1 チャンネルを優先的に使用することを勧めます、チャック爪は 2 経路の RS485 をいずれも使用可能です。そして、2 経路 RS485 チャ

ンネルの内、一つしかトルクセンサーとして利用できません。

RS485 チャンネル通信パラメーターの設定

RS485 チャンネルのモードにかかわらず、ボーレート（最大 2250000 まで対応）、データビット（8/9 まで対応）、ストップビット（1/2 まで対応）、照準ビット（奇数/偶数/なしまで対応）など、RS485 チャンネルのシリアル通信パラメータを設定する必要があります。チャンネルモードが Modbus RTU に設定されている場合は、Modbus スレーブノード ID を追加で指定する必要があります。

注：通信パラメーターの設定エラーは、TIO+ が外部装置との正常通信の支障の原因になります。

③トルクセンサーの使用

現在 TIO は、SRI、LONEKING 製のトルクセンサーなど、一部カスタマイズ可能なトルクセンサーにしか対応していません。トルクセンサーの利用には、下記の設定手順が必要とします：

- トルクセンサーを取り付け、トルクセンサーの接続端子を TIO エンドコネクタに接続します。
- TIO 電源出力設定、正確な電源出力幅を設定し、電源出力を有効にします；
- ボーレート、データビット、ストップビット及び照準ビットなど、RS485 の通信パラメーターを設定します；
- RS485 チャンネルのモードをトルクセンサーモードに設定します；
- 対応する IO ピンを RS485 機能に多重化させます。

上記手順の完了後に、TIO と外部トルクセンサーとの正常通信が実現できます。ただし、ロボットコントローラにて TIO によって拡張された端末トルクセンサー機能を真に利用可能にするためには、下記設定が必要とします。

- トルクセンサー設定画面にて、利用する TIO+ RS485 チャンネルを選びます；
- トルクセンサーを有効にします。

トルクセンサーがなお有効にできない場合、ピン機能の多重化設定及び RS485 通信パラメーター設定を再チェックする必要があります。正常に有効にした後、初めてトルクセンサーが稼働状態となります。APP 画面では、トルクセンサーのステータス値をモニタリングし、力制御機能をジョブプログラムにて利用することができます。

【ハードウェアと通信】->【端末センサー】画面にて、図 3-46 に示した通り、イーサネットトルクセンサーと端末 TIO チャンネルトルクセンサーとの切替ができます。

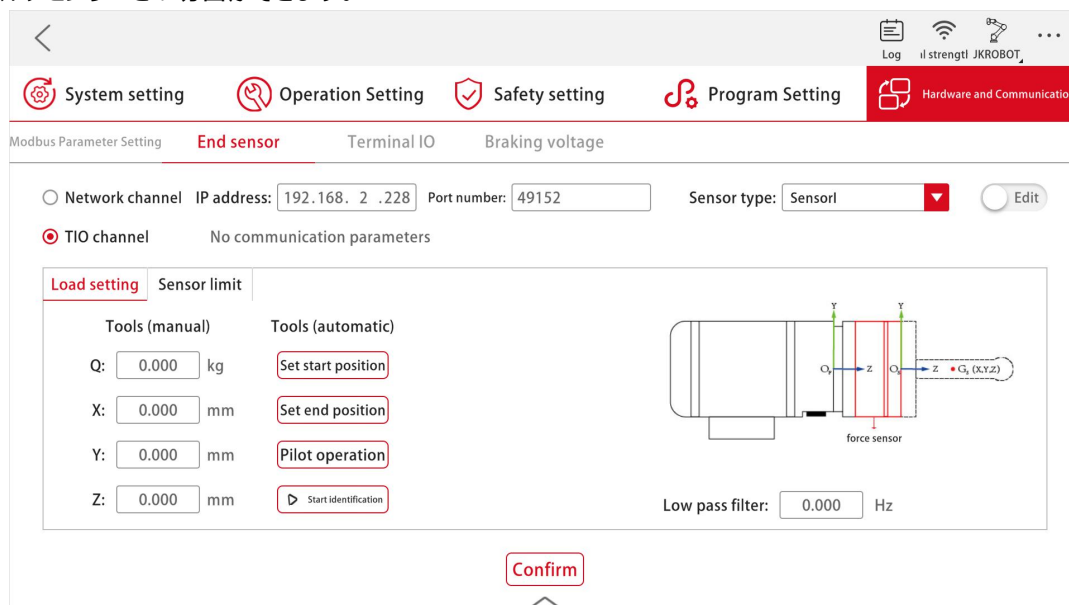


図 3-46 トルクセンサーのチャンネル使用選択

TIO チャンネルのトルクセンサーを利用する際、その通信パラメーターを RS485 チャンネル設定画面にて設定する必要がありますが、図 3-46 に示す画面では、使用する RS485 チャンネルの通信パラメータのみが表示されます。トルクセンサーに対して TIO の RS485 チャンネルが正しく設定されないまま、なお TIO チャンネルを選択すると、トルクセンサーが

正しく動作しません。トルクセンサーのオンオフ、トルクセンサーの負荷設定等、TIO チャンネルに基づいたトルクセンサーの使い方は、何れもイーサネット通信方式での使用方法と同様です。

現在のトルクセンサに TIO チャンネルが選択されており、現在通信可能な状態で接続されている場合、対応する TIO の RS485 チャンネル構成やピン構成を切り替えるとエラーが表示されるため、その前にトルクセンサの電源を切る必要があります。

④チャック爪の使用

現在対応しているのは、通信方式が Modbus RTU となるチャック爪のみです。チャック爪と TIO 端子を接続する前に、まず端子接続件の定義が正しく、正確な回線手順で接続されていることを確認します。詳しい通信方法は上記②RS485 の節に参考してください。

⑤信号量

JAKA コントローラには、ユーザーによる検索が可能な信号量が表示されます。ユーザーは、検索が必要なすべてのステータス量を事前に定義しておけば、その後、リフレッシュおよび検索操作によってステータス量の値を取得することができます。



図 3-47 信号量の定義

信号量の定義画面図に示した通り、各々の信号量の意味は以下の通りです：

信号量の表示：信号量の唯一の表示コード（Unicode と特殊文字に対応しません）、当該表示はその後にくリフレッシュ、取得及び削除操作にも利用できます；

RS485 チャンネル：信号発信元となる TIO の RS485 チャンネルが RS485-1 か RS485-2 かを指定します。

信号量タイプ：信号量のデータタイプ。当該パラメーターが Modbus 機能コードと対応して、01 のコイルレジスター、02 の分離入力

、03 の入力レジスター 04 のレジスター保持というものを以外は、一切対応していません。

信号量アドレス：即ち信号量に対応する Modbus レジスターのアドレスのことです。当該アドレスに RS485 チャンネル設定及び信号量タイプを組合せて利用すれば、指定の Modbus RTU スレーブレジスターへのアクセスが実現できます。

注意：信号量の定義は、TIO+関連の IO ピンが RS485 チャンネルに多重化され、且つ Modbus RTU モードに設定された状態で行う必要があり、モードやピンの多重化を変更すると、信号量の設定が失われます。

信号量のリフレッシュと検索：

信号量が定義されると、デバッグ画面を介してデバッグやモニタリングを行ったり、ジョブプログラムで直接使用することができます。どちらも心信号量の更新や検索のためのインターフェースを提供しています。リフレッシュ操作は、実際にコントローラと TIO+ の外付けチャック爪との間でデータのやり取りを開始します。コントローラと TIO+ 外部機器とのやりとりはリフレッシュ動作コマンドと非同期のため、リフレッシュ操作の後、確実にリフレッシュされた値を得るためには一定の間隔（推奨 100ms）が必要です。また、リフレッシュ操作では、リフレッシュ頻度を指定することができ、頻度が 0 の場合は 1 回限りのリフレッシュとして扱われます。頻度が 0 より大きい数値の場合は、下部の通信帯域と連動して最大化対応されます。

図 3-48 に示した通り、画面上の「リフレッシュ」アイコンを使って、信号量の値を手動で更新することができます。



図 3-48 信号量モニタリング画面

信号量の削除：

信号量はデバッグ画面にて手動で削除できます。信号量モニタリング画面にて、対応する信号量の右端にある削除アイコンをクリックすると、削除の確認画面が表示され、確認後に対応する信号量が図のように削除されます。

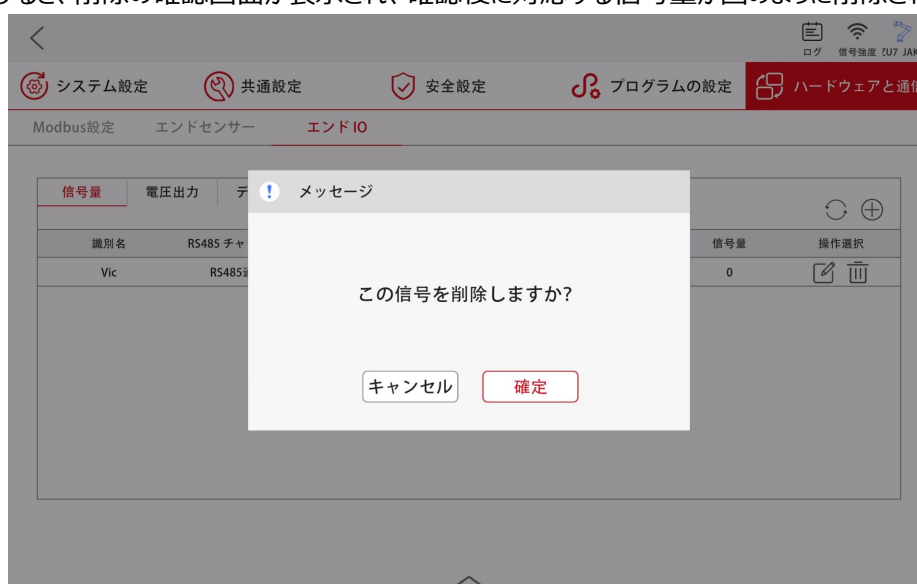


図 3-49 信号量の削除

⑥即時コマンドの発送

即時コマンドとは、主にコントローラから TIO 外部チャック爪に発信されるチャック爪の位置制御、スピード制御及び力制御などを含めた即時的な制御コマンドです。（注：ユーザーが即時コマンドデータを入力する場合は、コマンドデータを書き込むだけで、制御システム内部で自動的に CRC チェック値が追加されます。）

下図に示した通り、TIO 装置デバッグ画面にて、即時コマンドの発送を実現できます。16 進数のデータコマンドを入力後に送信をクリックすると送信完了です。



図 3-50 外部装置のコマンド制御

⑦ジョブプログラムにおける TIO+へのサポート

ジョブプログラムには、信号量をリフレッシュと検索するためのコマンドが備えています。信号量の定義、修正と削除はデバッグ画面にて手動追加する必要があります。そのほか、外部装置に対する即時的制御を実現するために、即時コマンドの発送コマンドにも対応できます。現在対応している UI コマンドは図に示した通りです。

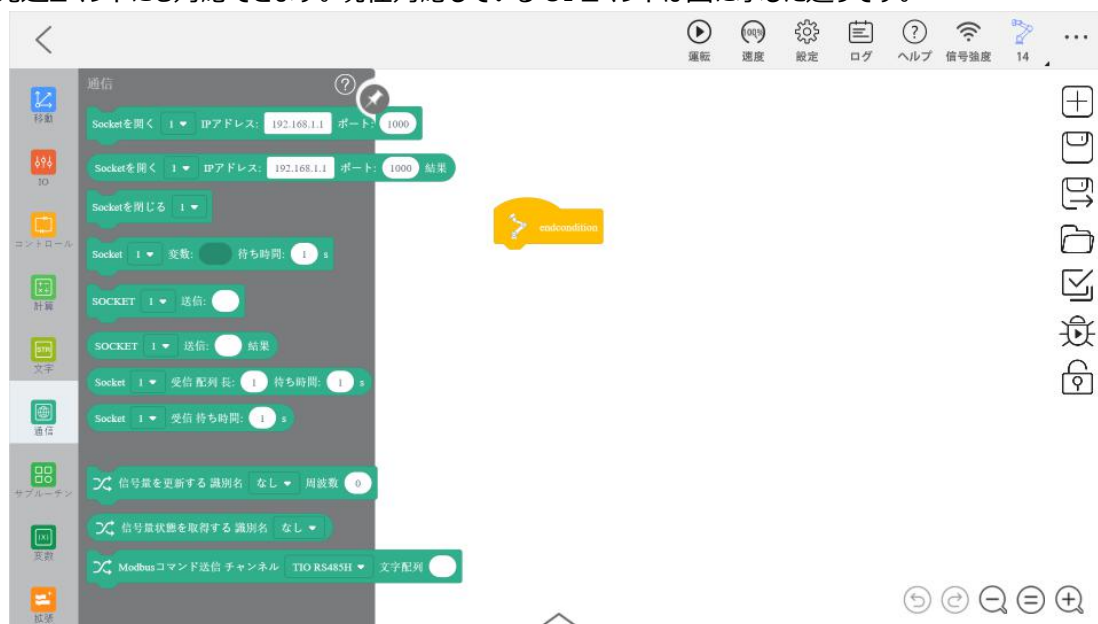


図 3-51 ジョブプログラムにおける TIO+へのサポート

3.1.5.4 ブレーキ電圧

注：本機能は MiniCab 制御盤バージョンのみに適用します。

MiniCab には、減速・制動時にロボットが発生する電気力を排出するための電圧制動回路が内蔵されています。ユーザーが外部電源を使用する場合は、過電圧保護の脱落やコントローラの破損を防ぐために設定する必要があります。ブレーキ電圧を設定する際、以下のルートに基づいてロボット本体の電源を落とす作業が必要となります：



図 3-52 ブレーキ電圧の設定経路

電圧設定値 V_{Brake} と入力電圧 V_{IN} の関係性は $V_{\text{Brake}} \geq (V_{\text{IN}} + 3)\text{V}$ の場合、よく利用する入力電圧タイプに基づいた推奨設定値とパワー電源種類との対照表は如下の通り：

表 3-53 入力電圧タイプに基づいた推奨設定値とパワー電源種類との対照表

電源種類	電圧 V_{IN}^1	ブレーキ抵抗起動電圧 V_{Brake}
48V モジュール電源	48V	51V
48V リチウム電池	54.6V	58V
24V モジュール電源 ²	24V	27V

- 注：
- 1、電圧値を代表値に変更；
 - 2、24V モジュール電源が Minicobot 専用電源です；
 - 3、設定された $V_{\text{Brake}} < (V_{\text{IN}} + 1)\text{V}$ の場合、内部ロジックが給電保護を行うとともに、APP に「本体電圧が電圧設定が異常しました」が表示されます。

3.2 I/O 情報の構成仕方

ロボットシステムの電気 I/O は JAKA Zu ソフト I/O モニタリング画面にて、表示と設定することができます。（注：I/O モジュールの編集・設定時には、ロボットの電源を切る必要があります）。I/O モジュールパネルは、制御盤 I/O とツール側 I/O に分かれます。下の「↑」矢印をクリックしてメニューバーを呼び出し、「I/O モニタリング」をクリックして I/O モジュール画面を開きます（メイン画面では「↑」矢印をクリックする必要はありません）。

注：

V2.1 版コントローラには、16 経路の入力と 16 経路の出力が配置されますが、APP 経由で V2.1 バージョンコントローラと接続した場合、I/O パネル画面に表示される DI、DO が各々 16 個になります。（V2.1 版コントローラは PNP タイプです）

MiniCab 版制御盤には、7 経路の NPN タイプ入力出力の多重化インターフェースが内蔵されますが、同一時刻には、一種のインターフェースタイプしか選択できません。

3.2.1 コントローラ IO

3.2.1.1 バージョン V1.0 コントローラ

バージョン V1.0 コントローラには、物理的な 8 経路入力、8 経路出力と 8 経路のアナログ入力が内蔵されますが、APP 経由でバージョン V1.0 コントローラと接続した場合、I/O パネル画面には制御盤の物理的な信号が表示されます。（バージョン V1.0 コントローラが NPN タイプです）図 3-54 に示した通り。

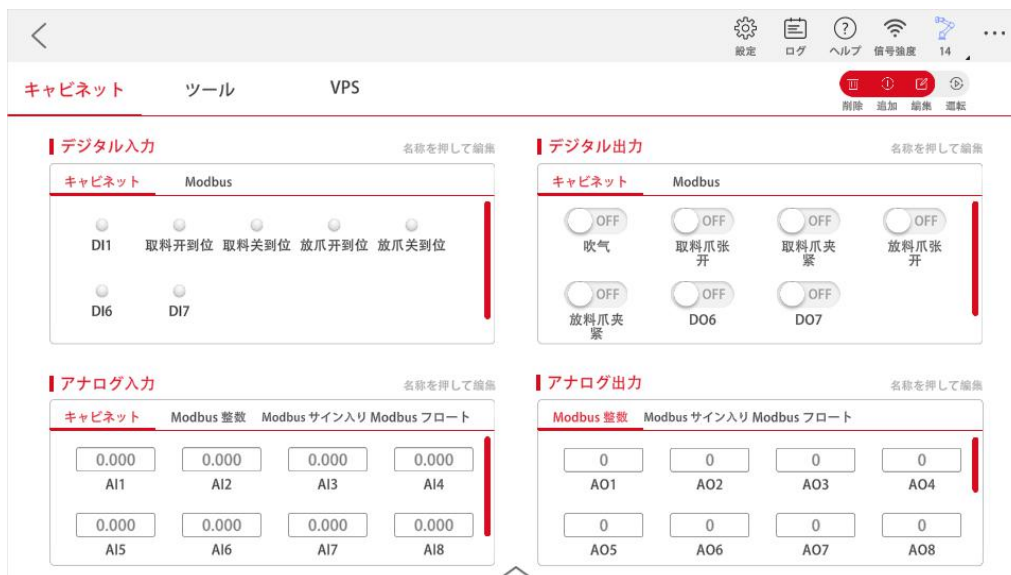


図 3-54 制御盤 I/O イメージ図

デジタル入力画面にてモニタリング制御盤中のデジタル入力状態をモニタリングすることができます。デジタル入力信号をクリックして、DI に対して対応する名前の編集と DI 機能の設定ができます。この DI に設定したい機能を機能選択ボックスでドロップダウンして選択し、クリックして確定すると、この DI 信号がトリガーされたときに、設定した機能も有効になります。（詳しい内容は 3.2.3 節に参照してください）コントローラのウィンドウにおける I/O 信号はロボットコントローラ中の物理的な I/O 信号で、Modbus ウィンドウにおける I/O 信号はロボットと外部装置との Modbus 通信後に、Modbus Address シート中のアドレスから読み取って制御を行った I/O 信号となります。図 3-55 に示した通り。

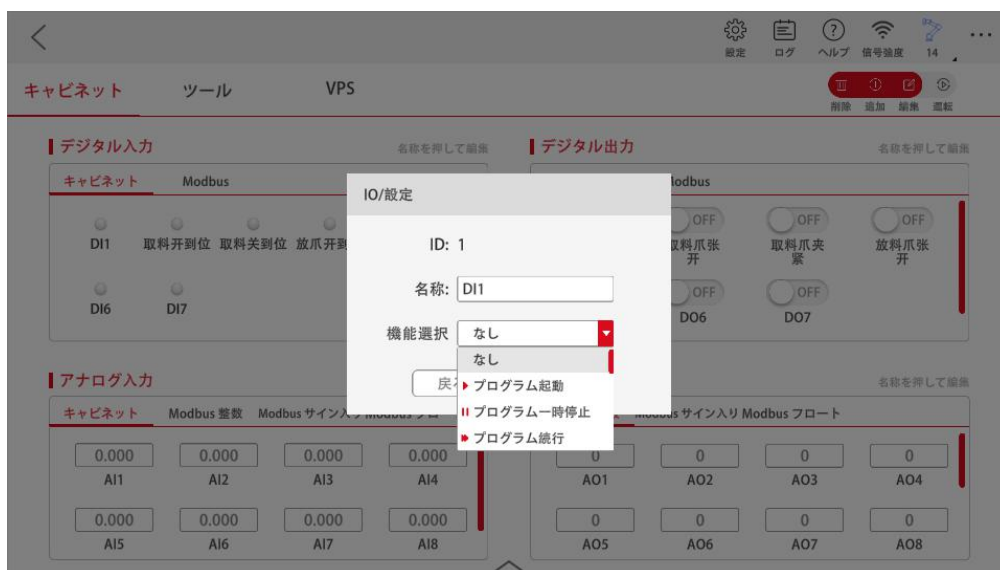


図 3-55 DI 設定イメージ図

デジタル出力画面にて、制御盤中のデジタル出力状態をモニタリングできます。デジタル入力信号をクリックして、DO に対して対応する名前の編集と DO 機能の設定ができます。機能選択セレクトボックスのドロップダウンから当該 DO と紐づけさせたシステムが予め定義したステータス量を設定して確認すると、当該 DO 信号がリアルタイムに紐づけされたシステムステータス量の状態を表示します。コントローラウィンドウ中の I/O 信号はロボットコントローラ中の物理的な I/O 信号で、Modbus ウィンドウ中の I/O 信号はロボットと外部装置との Modbus 通信後に、Modbus Address シート中のアドレスから読み取って制御を行った I/O 信号となります。図 3-56 に示した通り。

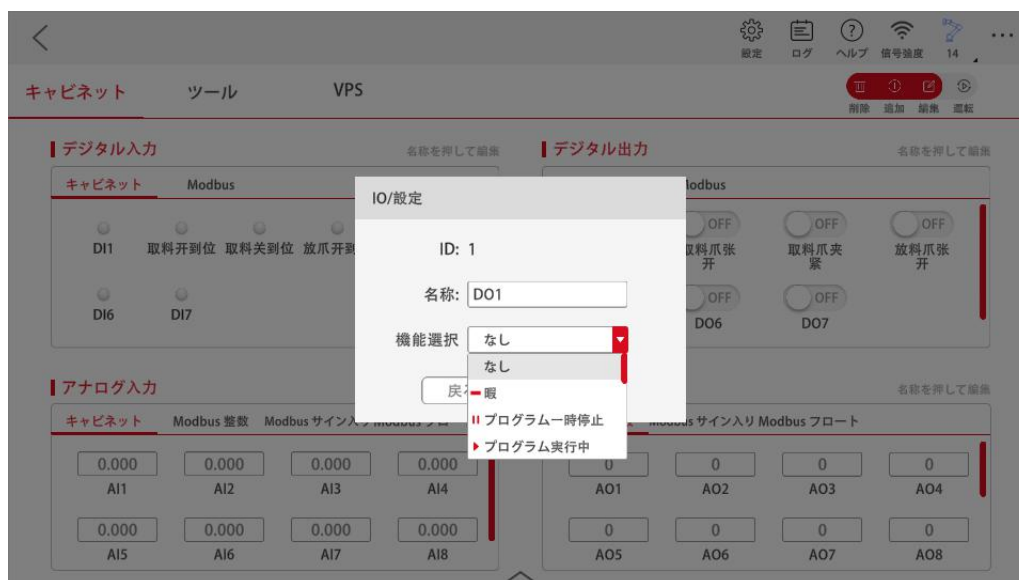


図 3-56 DO 設定イメージ図

アナログ入力画面では、アナログ信号量のモニタリングができます。I/O 画面にて、アナログ入力 AI に対する名前設定ができます。アナログ入力信号をクリックし、名前のテキストボックスをクリックしてカスタム名前を入力して確認すると、即発効します。（図 3-57 に示した通り）アナログ信号画面では、Modbus ウィンドウ中のアナログ信号はロボットと外部装置との Modbus 通信後に、Modbus Address シート中のアドレスから読み取って制御を行ったアナログ信号となります。ただし、Modbus アナログ信号の異なる場所でのアナログ信号がデータ種類上の差が存在するため、必ず使用されるデータ種類と対応した Modbus アナログ信号を採用するように注意してください。



図 3-57 AI 設定図

3.2.1.2 バージョン V2.1 コントローラ

バージョン V2.1 コントローラには、物理的なある 16 経路入力、16 経路出力と 2 経路アナログ信号が配置されますが、APP にてバージョン V2.1 コントローラと接続した場合、I/O パネル画面には、制御盤中の物理的な信号が表示されます。（バージョン V2.1 コントローラは PNP 型です）図 3-58 に示した通り。

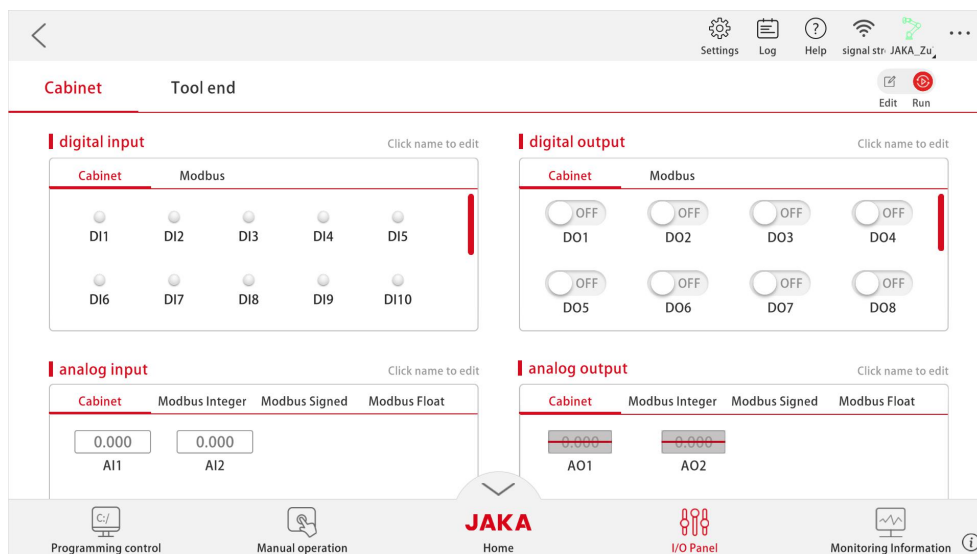


図 3-58 制御盤 I/O イメージ図

デジタル入力画面にてモニタリング制御盤中のデジタル入力状態をモニタリングすることができます。デジタル入力信号をクリックして、DI に対して対応する名前の編集と DI 機能の設定ができます。この DI に設定したい機能を機能選択ボックスでドロップダウンして選択し、クリックして確定すると、この DI 信号がトリガーされたときに、設定した機能も有効になります。（詳しい内容は 3.2.3 節に参考してください）コントローラウィンドウ中の IO 信号はロボットコントローラ中の物理的な I/O 信号で、Modbus ウィンドウ中の I/O 信号はロボットと外部装置との Modbus 通信後に、Modbus Address シート中のアドレスから読み取って制御を行った I/O 信号となります。図 3-59 に示した通り。

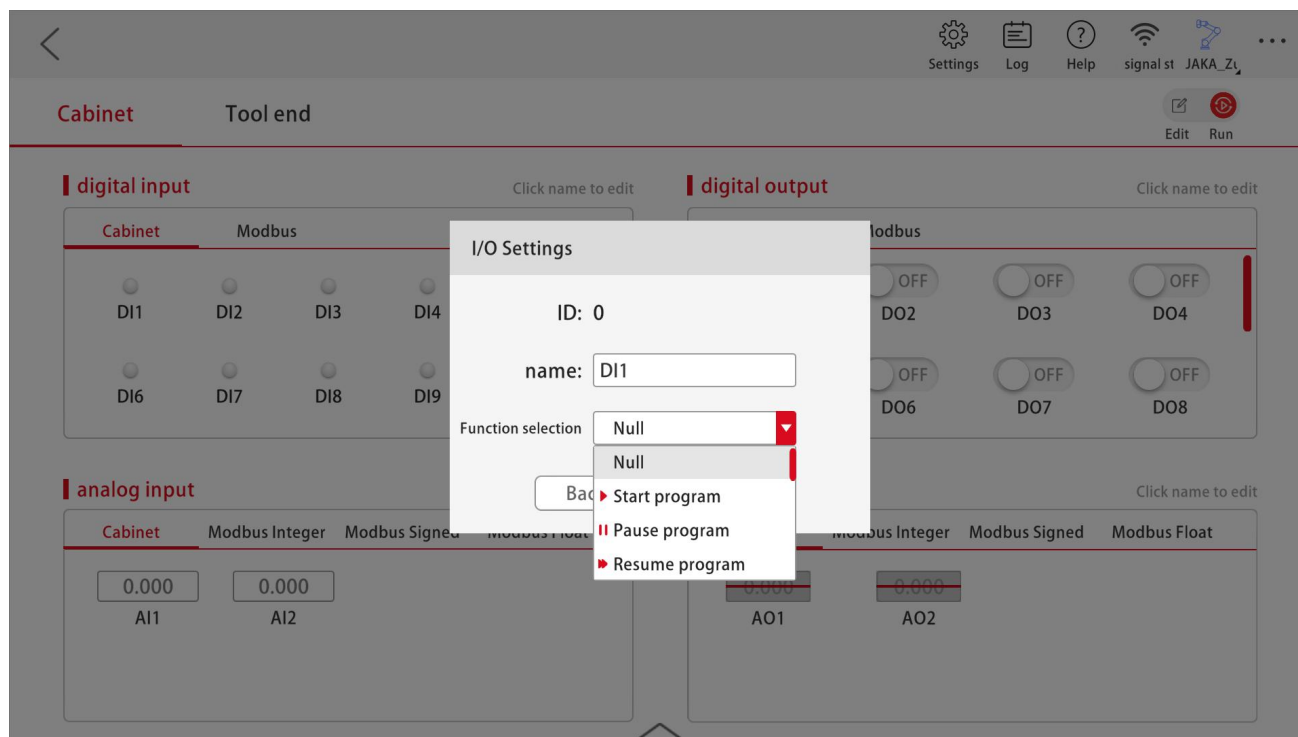


図 3-59 DI 設定イメージ図

デジタル出力画面にて、制御盤中のデジタル出力状態をモニタリングできます。デジタル入力信号をクリックして、DO に対して対応する名前の編集と DO 機能の設定ができます。機能選択セレクトボックスのドロップダウンから当該 DO と紐づけさせたシステムが予め定義したステータス量を設定して確認すると、当該 DO 信号がリアルタイムに紐づけされたシステムステータス量の状態を表示します。コントローラウィンドウ中の IO 信号はロボットコントローラ中の物理的な I/O 信号で、Modbus ウィンドウ中の I/O 信号はロボットと外部装置との Modbus 通信後に、Modbus Address シート中のアドレスから読み取って制御を行った I/O 信号となります。図 3-60 に示した通り。

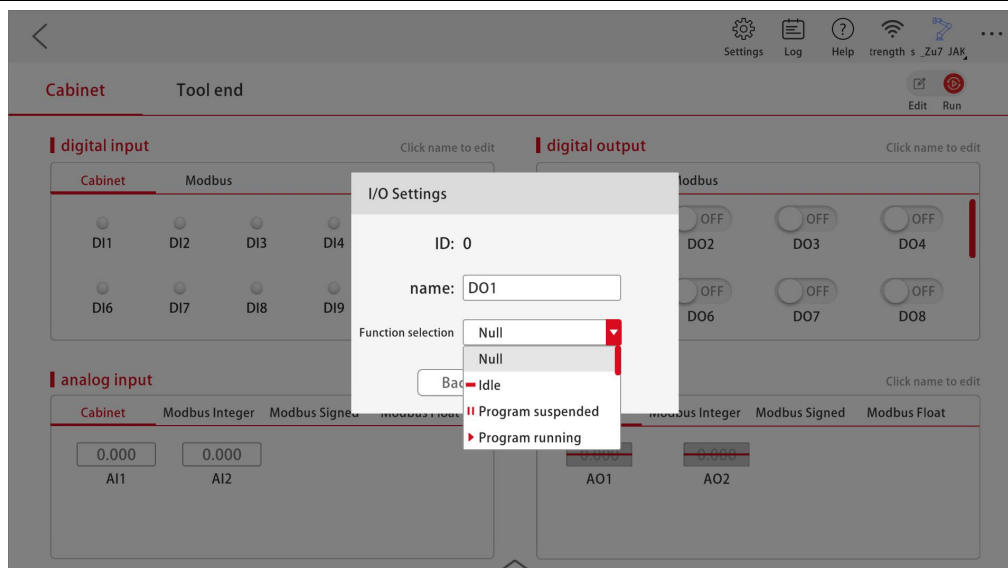


図 3-60 DO 設定イメージ図

アナログ入力/出力画面では、アナログ信号量のモニタリングができます。I/O 画面では、アナログ入力 AI に対する機能設定ができます。アナログ入力信号をクリックすると、当該信号の機能を選択したり、機能選択のドロップダウンから対応する機能を選んで、確認すると発効できます。現在、AI 信号が対応できる機能は：電圧入力、電圧出力、電流入力と電流出力となります。（図 3-61 に示した通り）；アナログ信号をクリックすると、当該アナログ信号に対する名前設定と数値設定ができるほか、同一画面にてアナログポートの設定（図 3-62 に示した通り）も実現できます。アナログ信号画面にて、Modbus ウィンドウ中のアナログ信号はロボットと外部装置との Modbus 通信後に、Modbus Address シート中のアドレスから読み取って制御を行ったアナログ信号となります。ただし、Modbus アナログ信号の異なる場所でのアナログ信号がデータ種類上の差が存在するため、必ず使用されるデータ種類と対応した Modbus アナログ信号を採用するように注意してください。

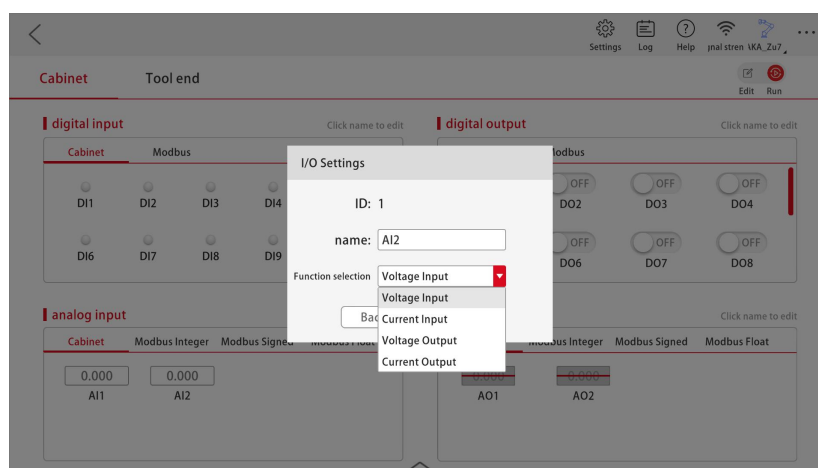


図 3-61 AI 設定図

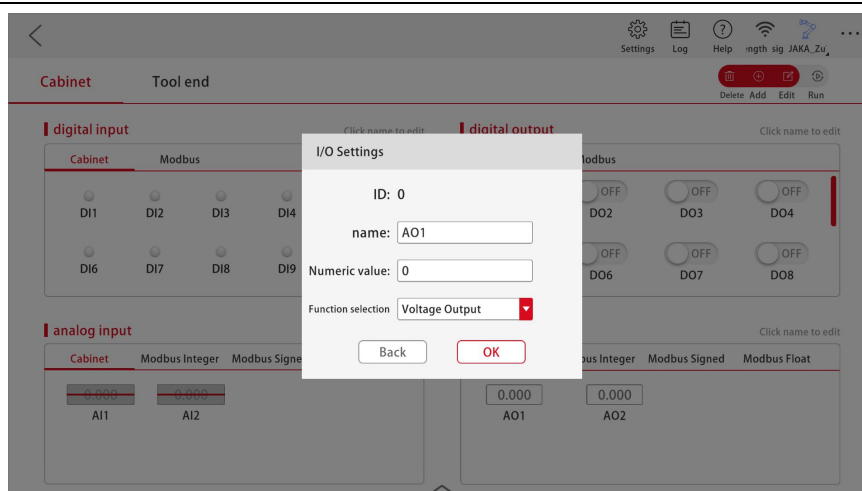


図 3-62 AO 設定図

3.2.1.3 MiniCab バージョンコントローラ

MiniCab バージョンのコントローラは、物理的な 7 経路のデジタル信号を持っています。APP 経由で MiniCab バージョンのコントローラを接続すると、I/O パネル画面に制御盤の物理的な信号が表示されます。（MiniCab バージョンコントローラは NPN 型です）図 3-63 に示した通り。

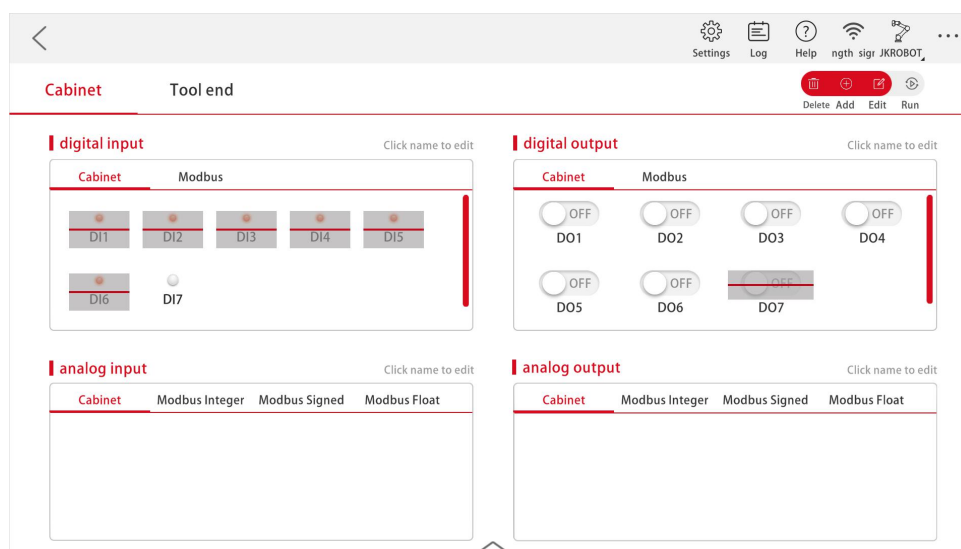


図 3-63 制御盤 I/O イメージ図

デジタル入力画面にてモニタリング制御盤中のデジタル入力状態をモニタリングすることができます。デジタル入力信号をクリックすると、対応する名前を編集したり、種類を選択したり、DI の機能を設定することができます。タイプの選択では、現在のインターフェースをデジタル入力として定義するか、デジタル出力として定義するかを選択できます。この DI に設定したい機能を機能選択ボックスでドロップダウンして選択し、クリックして確定すると、この DI 信号がトリガーされたときに、設定した機能も有効になります。（詳しい内容は 3.2.3 節に参照してください）コントローラのウィンドウにおける I/O 信号はロボットコントローラ中の物理的な I/O 信号で、Modbus ウィンドウにおける I/O 信号はロボットと外部装置との Modbus 通信後に、Modbus Address シート中のアドレスから読み取って制御を行った I/O 信号となります。図 3-64 に示した通り。

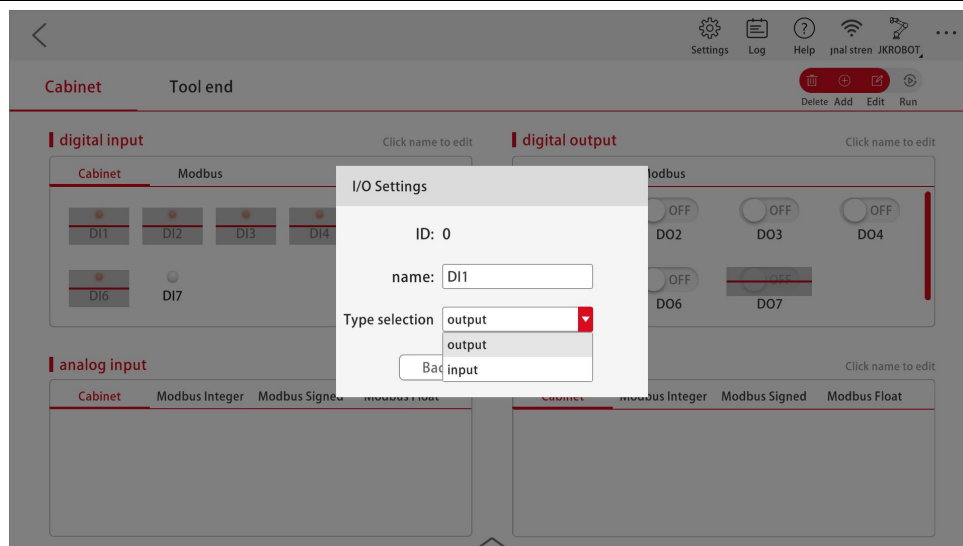


図 3-64 DI 設定イメージ図

デジタル出力画面にて、制御盤中のデジタル出力状態をモニタリングできます。デジタル出力信号をクリックすると、DO に対して名前を編集したり、種類を選択したり、機能を設定することができます。タイプの選択では、現在のインターフェースをデジタル入力として定義するか、デジタル出力として定義するかを選択できます。機能選択セレクトボックスのドロップダウンから当該 DO と紐づけさせたシステムが予め定義したステータス量を設定して確認すると、当該 DO 信号がリアルタイムに紐づけされたシステムステータス量の状態を表示します。コントローラウィンドウ中の I/O 信号はロボットコントローラ中の物理的な I/O 信号で、Modbus ウィンドウ中の I/O 信号はロボットと外部装置との Modbus 通信後に、Modbus Address シート中のアドレスから読み取って制御を行った I/O 信号となります。図 3-65 に示した通り。

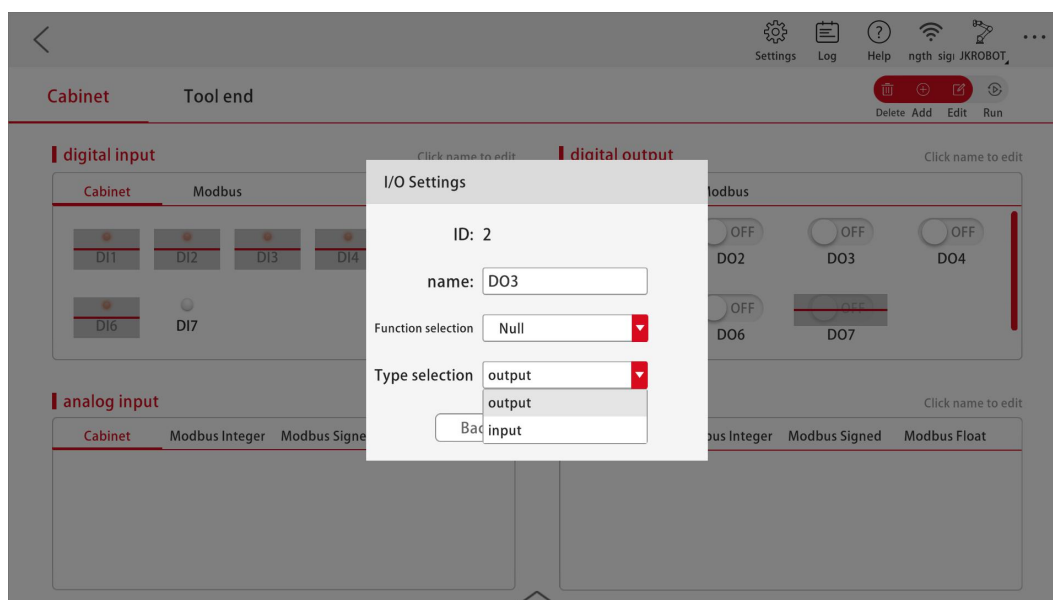


図 3-65 DO 設定イメージ図

3.2.2 ツール側 IO

ロボット本体端末ツール TIO は、V2 バージョンと V3 バージョンという 2 つのバージョンを持っています。V3 バージョンを使用する場合、端末 IO の設定は、設定画面のハードウェアと通信の画面で行います。端末ツール TIO のバージョンが異なる場合、IO パネルにおけるツール端末 IO によっても、以下のように異なる操作が可能です。

3.2.2.1 バージョン V2 ツール側 IO

ロボット本体バージョン V2 端末（ツール側）には、2 経路のデジタル入力、2 経路のデジタル出力と 1 経路のアナログ電圧入力配置されています。図 3-66 に示した通り。

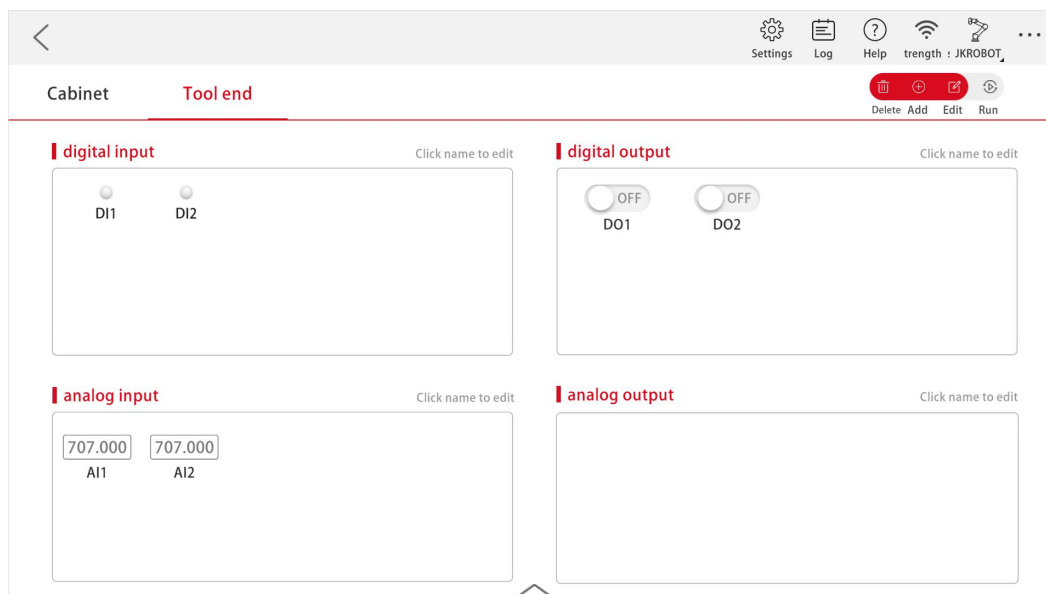


図 3-66 ツール側 I/O イメージ図

デジタル入力画面では、ツール側中のデジタル入力状態をモニタリングすることができます。デジタル入力信号をクリックして、DI に対して対応する名前の編集と DI 機能の設定ができます。この DI に設定したい機能を機能選択ボックスでドロップダウンして選択し、クリックして確定すると、この DI 信号がトリガーされたときに、設定した機能も有効になります。図 3-67 に示した通り。



図 3-67 DI 設定イメージ図

デジタル出力画面では、ツール側でのデジタル出力状態をモニタリングすることができます。デジタル入力信号をクリックして、DO に対して対応する名前の編集と DO 機能の設定ができます。機能選択セレクトボックスのドロップダウンから当該 DO と紐づけさせたシステムが予め定義したステータス量を設定して確認すると、当該 DO 信号がリアルタイムに紐づけされたシステムステータス量の状態を表示します。図 3-68 に示した通り。



図 3-68 DO 設定イメージ図

アナログ入力画面では、アナログ電圧信号をモニタリングすることができます。アナログ入力信号をクリックすると、当該信号の名前を変更（図 3-69 に示した通り）することができます；



図 3-69 AI 設定イメージ図

3.2.2.2 バージョン V3 ツール側 I/O

ロボット本体バージョン V 3 端末（ツール側）は、2 経路のデジタル入力、2 経路のデジタル出力及び 2 経路のアナログ入力に対応できます。その内、デジタル入力の 2 経路は高速 RS485 チャンネルとして多重化でき、アナログ入力の 2 経路は低速 RS485 チャンネルとして多重化できます。同時に、設定可能な電圧出力（12V/24V/0V）にも対応し、外部拡張装置への給電として利用できます。高度な設定など、詳しくは 3.1.5.3 の節に参照してください。

デジタル入力設定

TIO 機能における 2 経路 DI を、NPN 型入力か PNP 型入力の何れかに設定することができます。両方の DI は、TIO の電源異常の後、デフォルトで NPN 入力として設定されます。【主メニュー】->【IO モニタリング】->【ツール側】->【デジタル入力】画面に表示されるピン名をクリックすると、下図 3-70 に示した通り IO 設定ダイアログ画面が表示されますので、モード設定のドロップダウンにて対応ピンの入力タイプを設定することができます。

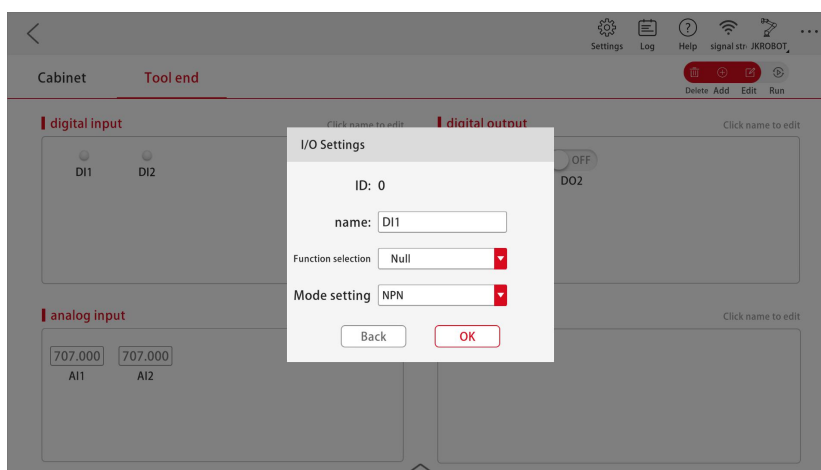


図 3-70 TIO デジタル入力ピンのモード設定

デジタル出力設定

TIO 機能での 2 経路 DO は、RS485 チャンネルピンとして多重化できます。それを DO として使用する場合は、NPN 出力、PNP 出力、プッシュプル出力など、異なる出力モードに設定することができます。2 経路の DO は、デフォルトでは NPN 出力として構成されます。【主メニュー】->【IO モニタリング】->【ツール側】->【デジタル出力】画面に表示されるピン名をクリックすると、下図 3-71 に示した通り IO 設定ダイアログ画面が表示されますので、モード設定のドロップダウンにて対応ピンの入力タイプを設定することができます。

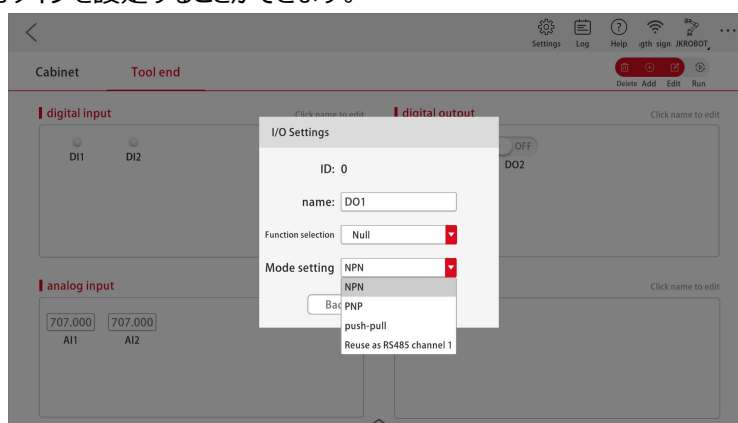


図 3-71 TIO デジタル出力ピンのモード設定

注意：2 経路の DO を RS485 チャンネルとして使用する場合、必ず同時に RS485 チャンネル 1 モードに設定する必要があります。片方の DO のモードが RS485 チャンネル 1 に変更されると、もう一方も RS485 チャンネル 1 に設定されます。片方の DO が RS485 チャンネル 1 モードから他のモードに設定されると、もう一方は自動的に NPN 出力モードに設定されます。

ドに設定されます。

アナログ信号の設定

TIO 機能では、2 経路の多重化可能なアナログ入力チャンネルが配置されており、2 経路のアナログ入力ピンを RS485 チャンネル 2 のチャンネルとして多重化することができます。TIO 給電後は、デフォルトとしてアナログ入力に利用されます。在【主メニュー】->【IO モニタリング】->【ツール側】->【アナログ入力】画面に表示されるピン名をクリックすると、下図 3-72 に示した通り I/O 設定ダイアログ画面が表示されますので、モード設定のドロップダウンにて対応ピンの入力タイプを設定することができます。

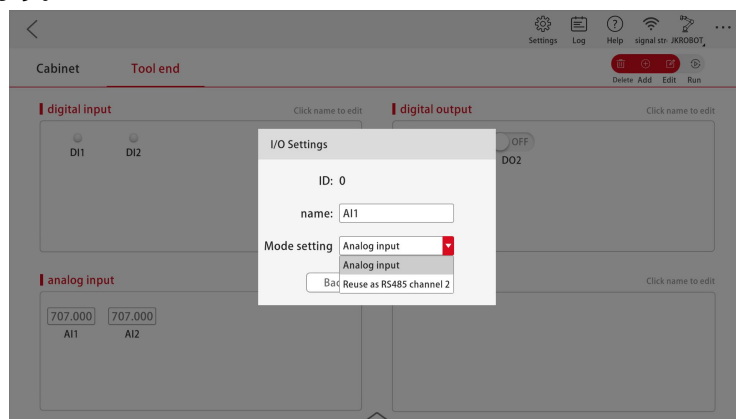


図 3-72 TIO アナログ入力ピンのモード設定

3.2.3 機能 IO

I/O 画面にて DI を設定できますので、機能選択ボックスでドロップダウンして当該 DI に設定したい機能を選択し、クリックして確定すると、この DI 信号がトリガーされたときに、設定した機能も有効になります。現在、DI 信号が対応できる機能は：プログラム開始、プログラム一時停止、プログラム継続、プログラム停止、電源オン、電源オフ、ロボット起動、ロボット停止、一級縮小モード、二級縮小モード、保護的停止、初期位置への復帰、故障の解除、ドラッグ機能の入力、ドラッグ機能の終了。表 3-73 に示した通り。機能のトリガー条件は、下表に示した通り：

表 3-73 機能トリガー方式リスト

機能名	トリガー方式
プログラム起動	上昇エッジ信号
一時停止プログラム	上昇エッジ信号
継続実行プログラム	上昇エッジ信号
停止プログラム	上昇エッジ信号
電源オン	上昇エッジ信号
電源オフ	上昇エッジ信号
ロボット起動	上昇エッジ信号
ロボット停止	上昇エッジ信号
一級縮小モード	ハイレベル信号
二級縮小モード	ハイレベル信号
保護的停止	ハイレベル信号
初期位置に戻る	ハイレベル信号
故障解除	上昇エッジ信号
ドラッグ開始	上昇エッジ信号
ドラッグ終了	上昇エッジ信号

注：

故障解除は衝突警報のみを対象とし、その他の異常警報は解除できません。。

二級縮小倍率が一級縮小倍率より低くなるように、設定、安全設定、保護システムにて倍率の設定を行います。

I/O 画面では、DO の機能を設定することができます。機能選択ボックスをドロップダウンして、DO と紐づけされるシステム定義済みのステータス量を設定し、クリックして確認すると、DO 信号は紐づけされたシステムステータス量の状態をリアルタイムで反映します。現在 DO 信号と紐づけられるステータスは以下のものが含まれます：アイドル、プログラム一次停止、プログラム実行、エラー、給電中、起動済み、運動中、静止、電源オン、システム緊急停止ボタン状態、ロボット縮小状態、システム保護的停止状態、安全位置など。

注：

アイドル状態とは、ロボットがプログラムを実行していない状態であり、ロボット本体の状態とは関係ありません。

エラー状態とは、ロボットが衝突警報をトリガーした状態を意味します。

運動中状態とは、ロボットが運動（プログラム実行状態、手動制御、二次開発制御運動等）となるとトリガーされ、プログラムの状態とは無関係です。

静止状態とは、ロボットが静止（プログラムの一次停止、プログラム未実行、プログラム終了、待機信号など）な状態のことを意味します、プログラムの状態とは無関係です。

電源投入状態とは、電動制御盤の電源投入状態を意味し、ロボット本体の状態とは関係ありません。

ロボット縮小状態とは、ロボットが二級縮小モード状態をトリガーしたか否かのことを意味します。

安全位置は、ロボットが安全設定におけるロボット姿勢にて設定された初期姿勢位置となるとトリガーされます。

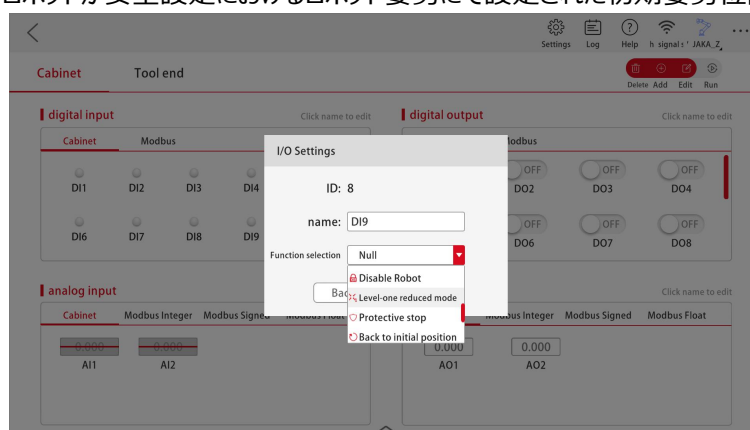


図 3-74 機能 DI イメージ図

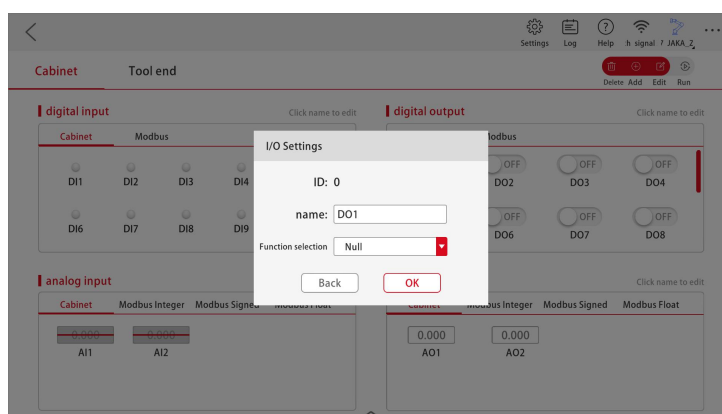


図 3-75 機能 DO イメージ図

3.2.4 I/O 増設拡張仕方

I/O パネルには、コントローラを Modbus 通信のクライアント/マスターにするためのダイナミック I/O 設定機能があります。この機能は I/O パネル画面の右上にあり、実行と編集という 2 つの状態があります。編集状態に切替えた場合：「+」をクリックすると、ダイナミック I/O 設定を行います。

Modbus-TCP と Modbus-RTU という 2 パターンがあり、Modbus と制御盤通信用インターフェースと接続するように、RTU モードは USB か Modbus RTU を介して接続し、TCP モードはイーサネット経由で接続するように設定します。設定手順は以下の通り：

3.2.5 Modbus-TCP



図 3-76 Modbus-TCP 画面

名前： Modbus-TCP 名前を付ける

IP アドレス： Modbus-TCP の IP アドレスを入力します。

装置 ID： 装置番号の記入

ポート番号： ポート番号の記入

そして、TCP の DI、DO、AI、AO のレジスタアドレスと数量を設定します。

確認をクリックした後、I/O モニタリング画面にて新規定義された Modbus モジュールが表示されます。

3.2.6 Modbus-RTU



図 3-78 Modbus-RTU 画面

名前： Modbus-RTU 名前を付ける

ボーレート：設定された Modbus-RTU のボーレート

停止位サイズ：所設定 Modbus-RTU 的停止位長さ

スレーブノード番号：設定された Modbus-RTU の局番号

データビット長さ：設定された Modbus-RTU のデータビット長さ

校正方式：設定された Modbus-RTU の校正方式

レジスターアドレスと数量：使用される Modbus-RTU のパラメーター情報に基づいて記入します。

確認をクリックした後、I/O モニタリング画面にて新規定義された Modbus モジュールが表示されます。

第 4 章 運動操作

4.1 運動制御

APP ホームページにて、画面切替タブコントロール中の手動操作をクリックし手動操作画面へ移ります。他の画面の場合は、画面下部中央の↑矢印をクリックして画面切替タブコントロールをプッシュアップし、手動操作をクリックして手動操作画面へ移ります。

助ちゃんロボットが JAKA Zu ソフトによってサポートしている 2 つの座標系カスタマイズ機能はそれぞれ以下のようなものがあります：ツール座標系とユーザー座標系。ツール座標系とユーザー座標系は、何れも設定機能における汎用設定オプションにてカスタマイズや使用ができます。詳しくは 3.1.2 の節を参照してください。手動操作画面では、ロボットモデルで 2 つの座標系を表現しています。（注：座標系を切替えてもモデルの座標が変わらず、モデルの座標系は参考のみとします。）図 4-1 に示した通り。

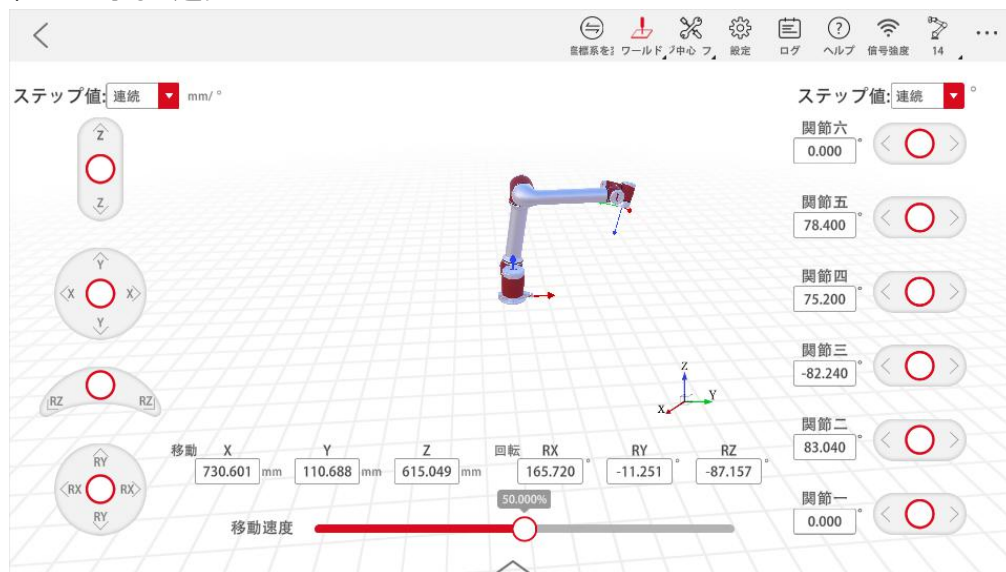


図 4-1 手動操作ウィンドウ図

4.1.1 運動参考座標系の切替仕方

手動操作画面では、「JOG 座標系切替」ボタンをクリックし、ユーザー座標系アイコンの色が赤になると、ユーザー座標系が現在使用されていることを表し、ツール座標系アイコンの色が赤になると、ツール座標系が現在使用されていることを意味します。⌛（ユーザー座標系は世界座標系としてデフォルト設定されています；ツール座標系は端末フランジ座標系としてデフォルト設定されています）。

4.1.2 ユーザー座標系の切替

ユーザー座標系アイコンの右下にある三角形をクリックし、ユーザー座標系の一覧をドロップダウンして、任意一つのユーザー座標系名称をクリックすると、当該ユーザー座標系に切替えることができます。図 4-2 に示した通り

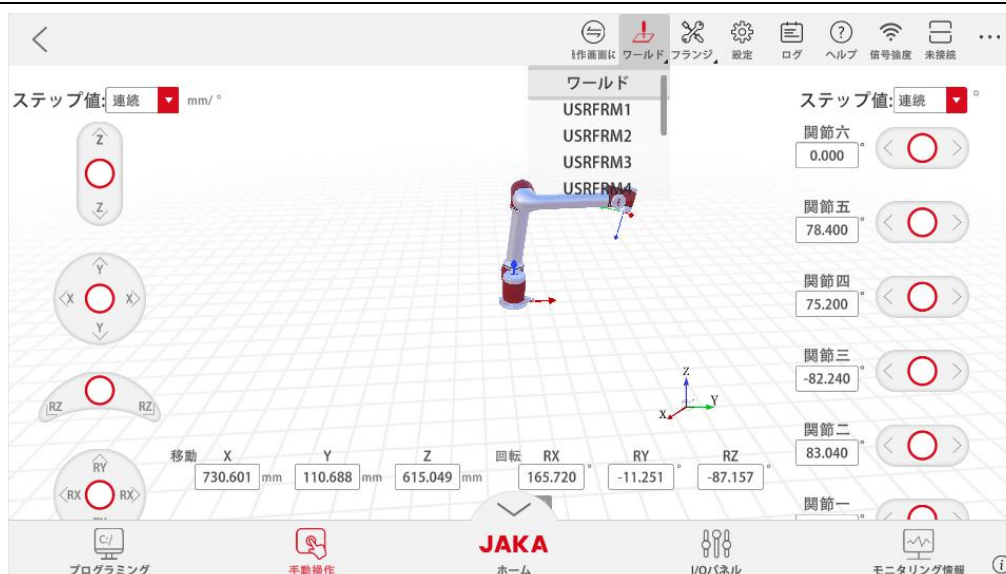


図 4-2 手動操作画面にてユーザー座標系へと切替える画面

4.1.3 ツール座標系の切替

ツール座標系アイコンの右下にある三角形をクリックし、ツール座標系の一覧をドロップダウンして、任意一つのツール座標系名称をクリックすると、当該ツール座標系に切替えることができます。図 4-3 に示した通り。

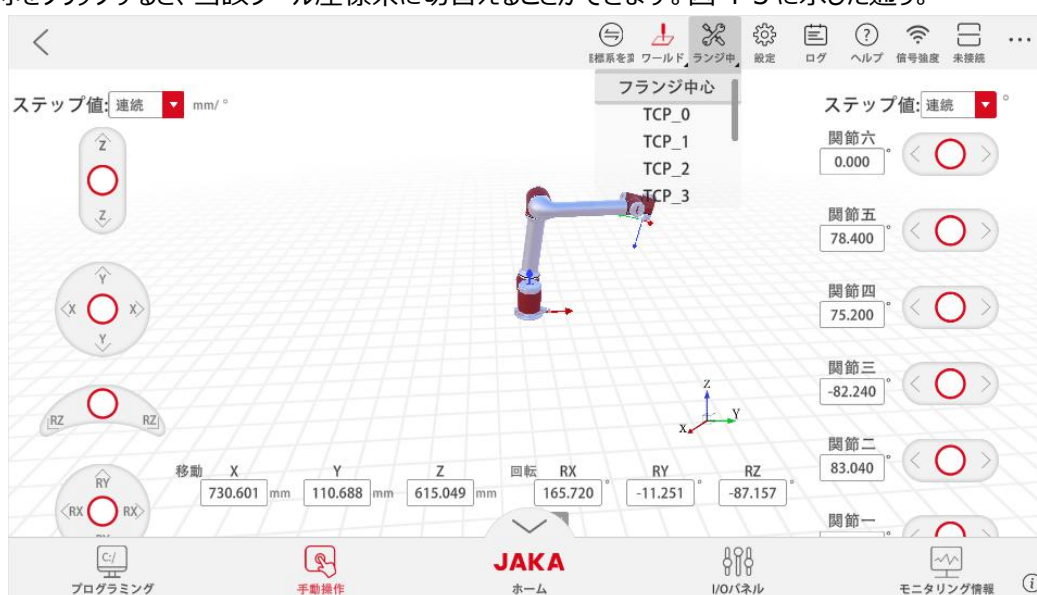


図 4-3 手動操作画面にてツール座標系へと切替える画面

4.1.4 ロボット運動の精確制御仕方

手動操作画面の各両側の上部には、ステップ値のオプションがあり、それぞれの下にあるバーチャルジョイスティックのステップ値を制御します。ステップ値を変えることで、手動操作ロボットの運動距離と運動角度をその都度に制御することができます。ステップ値が低いほど、ロボット運動もより精確になります。図 4-4 に示した通り

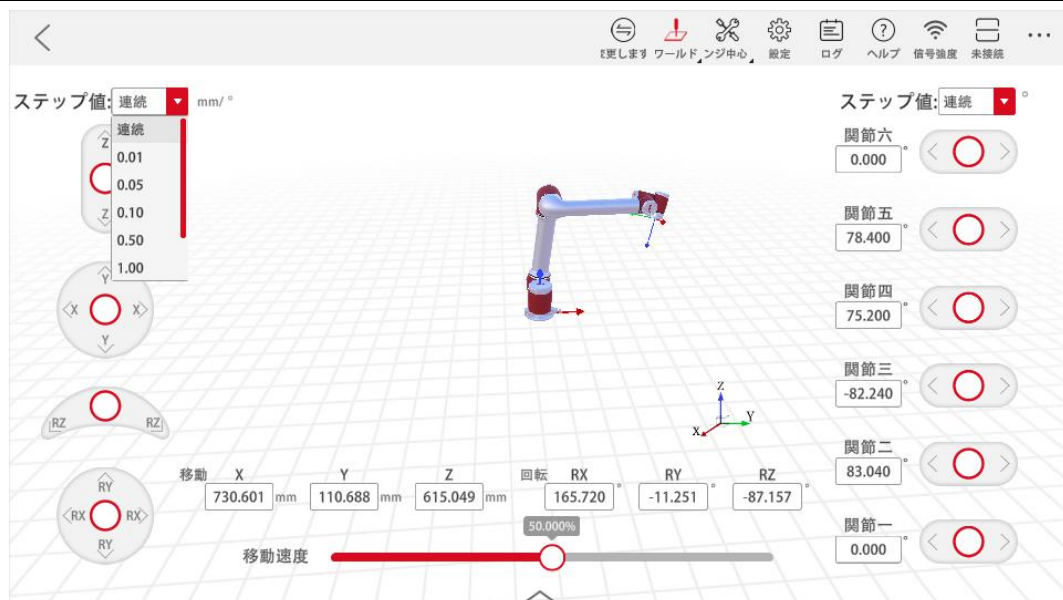


図 4-4 手動操作画面でのステップ値画面

4.1.5 運動スピードの制御

手動画面下方にある移動スピードバーをドロップすることで、ロボットの運動スピードを調整します。或いは、運動スピードバーの上にあるパーセントをクリックし、速度比を入力することもできます。

4.2 カルテシアン空間運動

空間運動とは、現在ロボットのツール座標系原点のことを意味します、デカルト空間やジョブ空間で運動しています。ユーザーは、ユーザー座標系かツール座標系の何れかで運動させるかを選ぶことができます。図 4-5 に示した通り。空間運動は各関節の協調運動となりますが、手動によるロボット空間運動を操作する方法は以下の通り：

画面左側のバーチャルスティックをスライドさせながら離さずキープすると、ロボットのツール座標系原点が現在ユーザー座標系においてそれ相応な空間運動を実行します。

バーチャルスティックを離れた場合、バーチャルスティックが自ずと原点に戻り、ロボットもすぐさま運動を停止します。

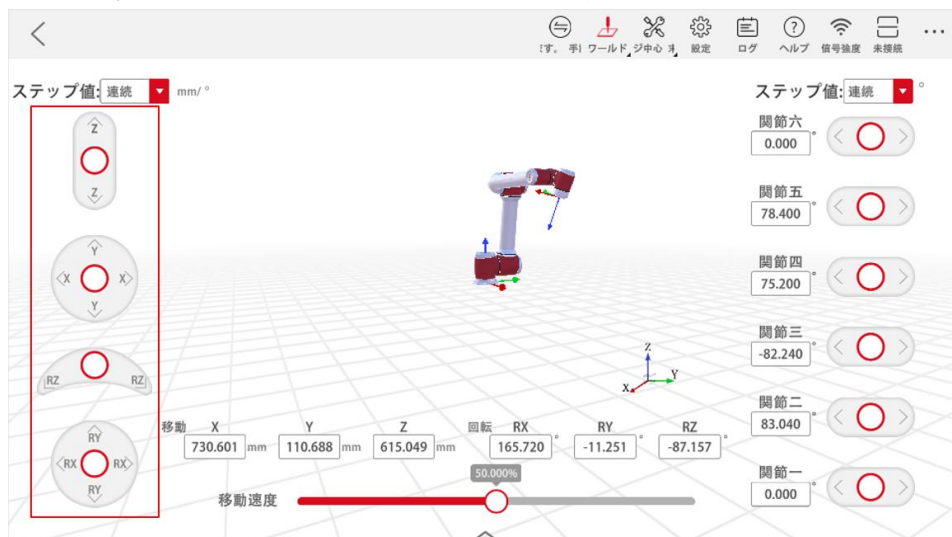


図 4-5 空間運動画面

4.3 関節運動

助ちゃんロボットが六つの関節によって構成されます（図 4-6 に示した通り）、これら関節を手動操作して互いに独立した動きを取らせることはいわゆる関節運動といいます、図 4-7 に示した通りとなります。手動にて単独関節の運動を操

作する方法は以下の通り：

画面右側のバーチャルスティックをスライドさせながら離さずキープすると、対応するロボットの関節は相応方向へと回転します。

バーチャルスティックを離した場合、バーチャルスティックが自ずと原点に戻り、ロボットもすぐさま運動を停止します。

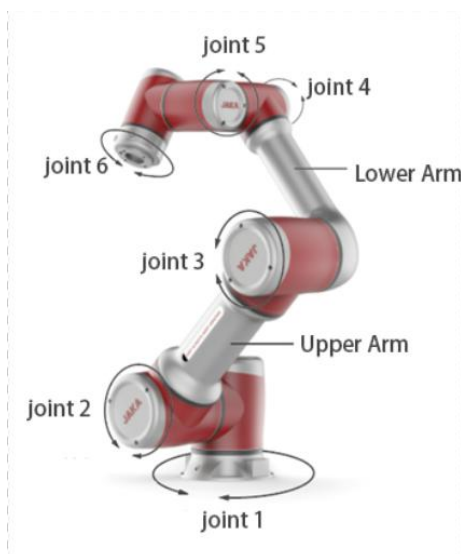


図 4-6 ロボット関節イメージ図

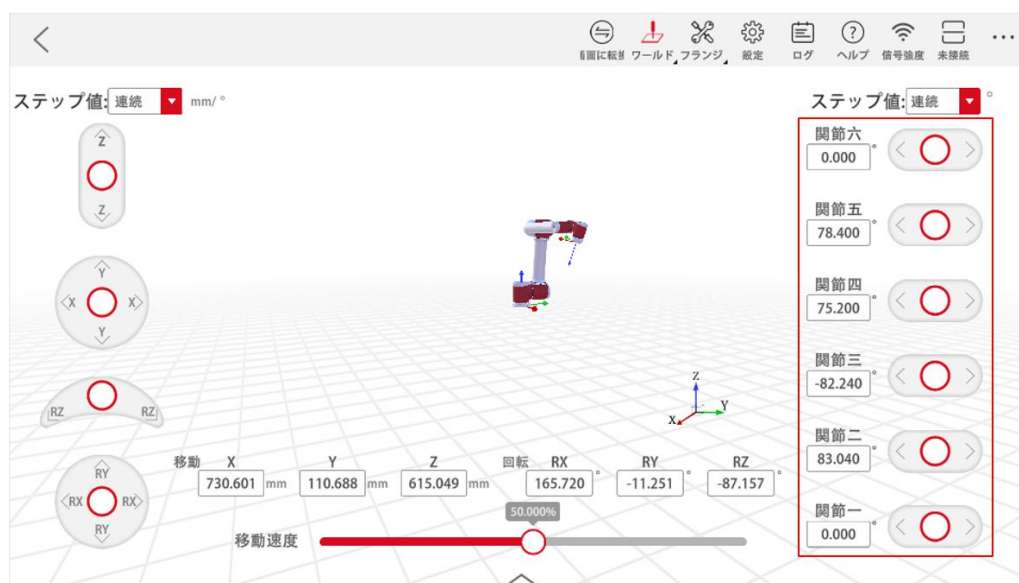


図 4-7 関節運動画面

4.4 位置運動

位置運動とは、ユーザーが手動でロボット運動を指定の位置へと操作することを意味します。その内、ユーザーはロボットの関節位置を指定したり、現在ツール座標系の原点が現在ユーザー座標系における空間位置を指定することができます。図 4-8 に示した通り。手動による位置運動を操作する方法は以下の通り：

手動操作画面にて任意一つ関節の情報や空間位置の情報ボックスをクリックして位置運動画面へと移ります。

(1) 指定した関節位置にロボットを運動させる必要がある場合は、関節位置エリアで、6つの関節の終了位置を入力し、長押ししてその位置に運動させると、ロボットは指定した位置に運動し、確定をクリックして位置運動画面を終了し、位置運動を完了します。

(2) 指定した空間位置にロボットを運動させる必要がある場合は、空間座標位置エリアで、終点の空間位置を入力し、クリックして関節位置を計算した後、長押ししてその点に運動させると、ロボットは指定した位置に運動し、確定をクリックして位置運動画面を終了し、位置運動を完了します。

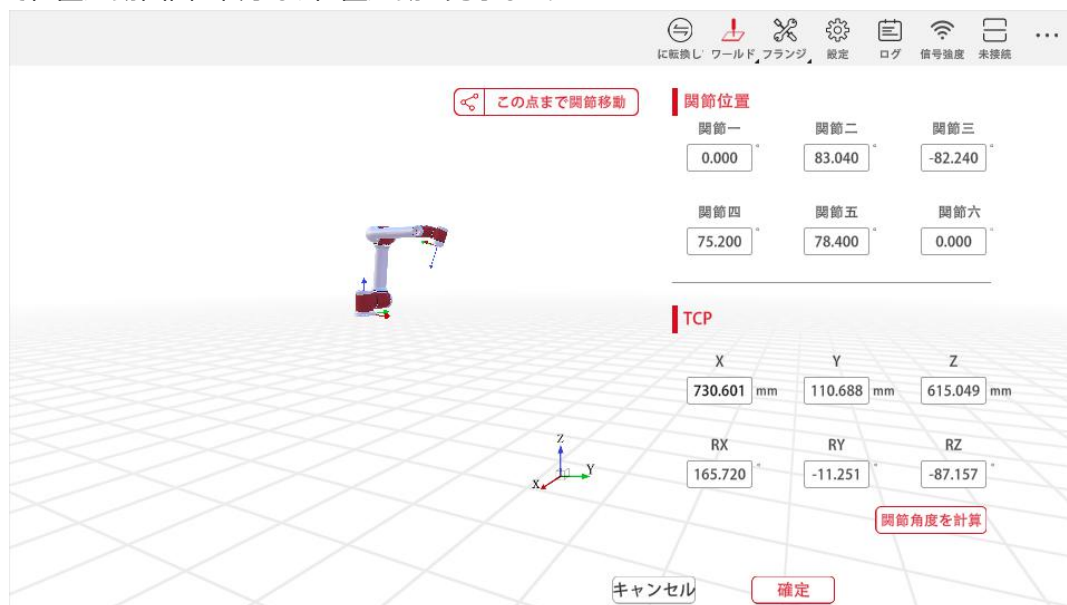


図 4-8 位置運動画面

第 5 章 スタートプログラミング

5.1 コマンド説明

JAKA Zu APP は、わずかなプログラミングの知識でも助ちゃんロボットに対するプログラミングができる利便性の高いプログラミング方法を提供しており、そのプログラミング方法は視覚的であるため、作業効率を大幅に向上させることができます。

5.1.1 プログラミングの新規作成



- (1) 意味：これより先ロボットが実行するプログラム
- (2) 使い方：実行したいコマンドを順に新規作成プログラミングの下に追加します；コマンドをクリックするとプログラム名前を修正することもできます。

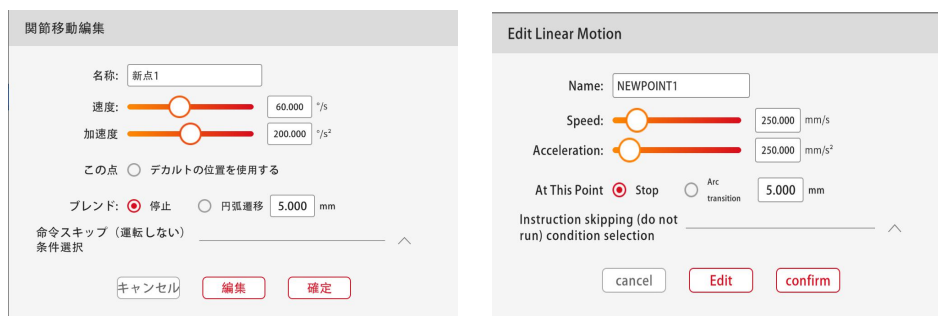
5.1.2 移動コマンド

5.1.2.1 関節運動或いは直線運動



- (1) 意味：終点を設定したり、変数を追加してロボットに関節運動をさせたり、端末に直線運動をさせたりします。
- (2) 使い方：コマンドのドロップダウンオプションにて、ロボットが目標ポイントへと移動させる運動形式（関節運動か直線運動）を選ぶことができます。

コマンドの設定画面を開くと、当該コマンドの名前、スピード、加スピードを編集したり；デカルト位置（関節運動）の使用選定したり；このポイントに停止させるか円弧遷移させたり；停止条件（ロボットが設定された停止条件に満足した場合、プログラムはこのコマンドをスキップする）を選んだり；ティーチングポイントを編集したりすることができます。



編集をクリックして手動ティーチング画面へ移り、ロボットを当該コマンドの目標位置まで運動させてから確定をクリックすると、ロボットの現在位置情報も当該コマンドの中に保存されます。コマンドにて保存済みロボット位置情報がある場合、「このポイントまで移動する」ボタンを長押しして、ロボットをこの目標位置へと運動させることができます。

注意：ロボットへの手動ティーチングを行う際、座標系が正しくセッティングされることを確認してください。間

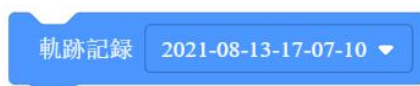
違った座標系は、ロボットの実質運動位置とアンマッチする原因になります。

関節運動を使用する際、デカルト位置オプションがチェックされていると、現在の運動で保存されたロボットのデカルト位置を使って、対応する関節角度値を解き、設定された位置まで関節運動を行います。

(3) 例：ロボット関節を初期位置に運動させてから、直線運動を行います。



5.1.2.2 運動軌跡



(1) 意味：JOG かドラッグの方法で、希望の目標ポイントと軌跡を採取して、軌跡を再現します。

(2) 使い方：コマンドのドロップダウンリストでは、使用する軌跡を選択することができます。コマンドをクリックすると、設定画面が表示され、このコマンドの速度と加速度を編集することができます。



(3) 例：ロボットが記録された軌跡に基づいて運動を始めます。



5.1.2.3 相対直線運動



(1) 意味：現在の姿勢に対して、ロボットが端末直線的運動を実行させるための位置増分を設定します。

(2) 使い方：コマンドをクリックすると設定画面が表示され、このコマンドの名前、速度、加速度を編集すること。参照座標系、ユーザー座標系、ツール座標系の選択、ここで停止するか円弧遷移させるかを選択すること。ロボットの端末相対的な変位量を入力すること。または、停止条件を選択（ロボットが停止条件で設定された状態を満たすと、プログラムはこのコマンドをスキップする）することなどがあります。

相对直線移動編集

名称:

参考座標系:

速度: mm/s

加速度: mm/s²

ブレンド: ☒ 停止 ☐ 円弧遷移 mm

X: mm Y: mm Z: mm

RX: ° RY: ° RZ: °

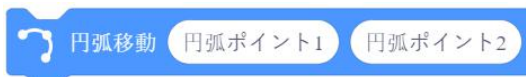
命令スキップ (運転しない) ^

条件選択

(3) 例：ロボットが初期位置へ運動させた後に、一部分の相对直線運動をさせます。



5.1.2.4 円弧運動



(1) 意味：三つのポイントを設定して、ロボット端末に円弧運動を実行させます。

(2) 使い方：コマンドをクリックすると設定画面が表示され、このコマンドの名前、速度、加速度を編集することができます。また、円弧の通過点と終点を教えることができます（円弧の始点は初期点です）。運動ターン数を入力することができます（この動きのターン数は円弧の初期点から計算され、ターン数が0の場合、実際の円弧の終点は設定された円弧の終点となります）。停止条件を選択することができます（プログラムは、ロボットが停止条件で設定された状態を満たすと、このコマンドをスキップします）。

円弧運動編集

速度: mm/s

加速度: mm/s²

ブレンド: ☒ 停止 ☐ 円弧遷移 mm

経過点設定:

運動圏数:

命令スキップ (運転しない) 条件選択

キャンセル 確定

(3) 例：ロボット端末に真円運動をさせます。

円弧運動の初期ポイント及び 2 つの経過ポイントを編集し、運動ターン数を 1 と入力します。



5.1.2.5 パレットウェイポイント



(1) 意味：パレットプログラムにおけるロボットが各パレットポイントに到達してからの運動終点を設定します。

(2) 使い方：コマンドをクリックすると設定画面が表示され、このコマンドの名前、速度、加速度の編集、このポイントで停止するか円弧遷移にするかの選択、停止条件の選択（停止条件で設定された状態になるとプログラムはこのコマンドをスキップする）、ポイントの位置の編集が可能です。

パレットのウェイポイント編集

名称:

速度: mm/s

加速度: mm/s²

ブレンド: ☒ 停止 ☐ 円弧遷移 mm

命令スキップ (運転しない) 条件選択

キャンセル 編集 確定

注意：このコマンドは、パレットフレーム内のみに発効できます。

例：軌跡の設定

ロボットが各パレットポイントまでに運動した後、垂直下方に 100mm を運動させます。



5.1.2.6 関節運動用のデカルト座標系

関節運動にはデカルト座標を使う ☒ オン ▼

- (1) 意味：デカルト座標系で記録したポイントを用いて関節運動を実行させます。
- (2) 使い方：関節運動の前にこのコマンドを使用することで、その後の関節動作は、デカルト座標系に記録された点で動くことになり、動作は関節動作の形のままですが、ロボットは自動的に逆解を行います。

5.1.2.7 設定全体スピード

グローバル速度をオン ☒ デフォルトの速度 ▼

- (1) 意味：全体スピードの設定を有効にします。
- (2) 使い方：リストをドロップダウンし、適当なスピード変数を選びます。全体スピード設定を有効にしてから、その後に続く直線運動と関節運動コマンドによる運動スピード、加速度は選定されたスピード変数と同様になります。

5.1.2.8 全体スピードの無効化

グローバル速度をオフ

- (1) 全体スピード設定の無効化
- (2) 使い方：このコマンドを実行した後に、全体スピード設定を無効化し、その後に続く直線運動と関節運動が元に設定した運動スピード、加速度に戻ります。

5.1.2.9 MoveZ

MoveZ 新規ポイント

- (1) 意味：ロボットが同一姿勢のままに Z の字型軌跡を運動させます。
- (2) 使い方：コマンドをクリックすると設定画面が表示され、運動開始位置と終了位置（Y 値の変化のみを監視し、+Y 方向への移動のみ可能）、Z の字の幅（Z 軌道の幅）、移動速度、Z の字の密度（設定しない場合はデフォルトの 45°とする）、円弧遷移値またはピーク一時停止時間（どちらか一方を選択）を編集することができます。

編集MoveZ

名称

開始位置

終了位置

任意位置

Zタイプの幅 mm 移動速度 mm/s

☐ Z型密度

☒ 円弧遷移 mm

☐ ピークタイムアウト時間 +X側点 ms -X側点 ms

5.1.2.10 MoveZS

MoveZS

(1) 意味：ロボットが RY 回転するとともに、+Y 方向へ移動させます。

(2) 使い方：コマンドをクリックすると設定画面が表示され、運動開始位置と終了位置（Y 値の変化のみを監視し、+Y 方向への移動のみ可能）、Z の字の幅（Z 軌道の幅）、移動速度、Z の字の密度（同側ピーク値との距離。一つの運動ターンを表し、密度はロボットの運動サイクルを決めます）、回転角度（設定しない場合はデフォルトの 45°とします）円弧遷移値かピーク値一時停止時間（どちらか一方を選択）を編集することができます。

編集MoveZS

名称

開始位置

終了位置

Zタイプの幅 mm 移動速度 mm/s

☐ Z型密度

☐ 回転角度

☒ 円弧遷移 mm

☐ ピークタイムアウト時間 +X側点 ms -X側点 ms

5.1.2.11 ロボット関係パラメーターの取得

(1) 意味：ロボット現在関節位置、ツール側センター位置、フランジセンター位置と姿勢、端末負荷、端末力、衝突検出度とシステム時間を取得します。

(2) 使い方：上位コンピューターへ送信するか変数値の代入ができます。各パラメーターのデータ種類は付録一に参照してください。

(3) 例：現在角度値の読取

上位コンピューターが現在ロボットの一から六番関節までの角度値を受信します。



5.1.2.11 ロボットユーザー座標系の設定或いはツール座標系の設定の取得



- (1) 意味：ロボットユーザー座標系の設定或いはツール座標系の設定を取得します。
- (2) 使い方：上位コンピューターへ送信するか変数値の代入ができます。
- (3) 例：現在ツール側座標値を読み取ります。

上位コンピューターが現在ロボットのツール側座標値を受信します。



5.1.3 IO コマンド

5.1.3.1 デジタル出力 () () を (オン/オフ) に設定



- (1) 意味：即時コマンドで、DO のオンとオフを設定します。
- (2) 使い方：ロボットがこのコマンドを実行する際、対応する DO 信号を設定された状態に切替えます。

5.1.3.2 運動中にデジタル出力 () () を (オン/オフ) に設定



- (1) 意味：非即時コマンドで、DO のオンとオフを設定します。
- (2) 使い方：円弧遷移を伴う直線運動コマンドの後にこのコマンドを使用すると、ロボットが直線運動前に対応する DO の状態をコマンドで設定した状態に切り替えます。

5.1.3.3 デジタル入力 () () を (開/关) に設定する待受け時間 ()



- (1) 意味：DI 状態がオン/オフになるまで、() s を待機させます。
- (2) 使い方：() 秒以内に、IO 状態がコマンドによって設定された状態になると、プログラムが引き続き実行されています。コマンドの設定した待受け時間をオーバーしても、プログラムが引き続き実行されています。(設定した待受け時間が 0s の場合、ロボットは IO が条件満足するまで待ち続けます。)

5.1.3.4 アナログ出力 () () を () に設定。



- (1) 意味：即時コマンドで、アナログ出力の値を設定します。
- (2) 使い方：AO 的値を設定します。

5.1.3.5 運動値としてアナログ出力 () () を () に設定

🔄 動作中にアナログ出力をセット コントローラ ▼ AOI ▼ が 1 である

(1) 意味：非即時コマンドで、AO の値を設定します。

(2) 使い方：円弧遷移を伴う直線運動コマンドの後にこのコマンドを使用すると、ロボットが直線運動前に対応する AO 値をコマンドで設定した数値に変更します。

5.1.3.6 デジタル入力 () () を (オン/オフ) に設定

🔄 デジタル入力 コントローラ ▼ DI1 ▼ が On ▼ である

(1) 意味：対応する DI の状態を取得します。

(2) 使い方：プログラムにおける判断条件としても利用できます。IO 状態とコマンドでの設定が一致した場合、真実とします。

5.1.3.7 デジタル出力 () () を (オン/オフ) に設定

🔄 デジタル出力 コントローラ ▼ 吹気 ▼ が On ▼ である

(1) 意味：対応する DI の状態を取得します。

(2) 使い方：プログラムにおける判断条件としても利用できます。IO 状態とコマンドでの設定が一致した場合、真実とします。

5.1.3.8 アナログ出力 () () の取得

🔄 アナログ出力 コントローラ ▼ AOI ▼

(1) 意味：現在ロボットの AO 値を取得します。

(2) 使い方：変数値代入の方法で、アナログ出力の値を変数に代入することができます。

5.1.3.9 アナログ入力 () () の取得

🔄 アナログ入力 コントローラ ▼ AII ▼

(1) 意味：現在ロボットの AI 値を取得します。

(2) 使い方：変数値代入の方法で、アナログ出力の値を変数に代入することができます。

5.1.4 制御コマンド

5.1.4.1tcp 設定

TCP設定 新規TCP

(1) 意味：TCP 座標系を設定します。

使い方：コマンドをクリックして設定画面へ移ると、座標系の名前と座標値を編集します。この座標系は当該プログラムでのみ使用可能です。（使用しない場合、ロボットはデフォルトとして端末フランジ座標系を使用します。）

工具座標系編集

座標名称: 新規TCP

X 0	RX 0
Y 0	RY 0
Z 0	RZ 0

キャンセル 確定

5.1.4.2 tcp () の設定

TCP設定 フランジ中心 ▼

(1) 意味：設定された TCP を選択します。

5.1.4.3 ユーザー座標系の設定

ユーザー座標系を設定 新規UCS

(1) 意味：ユーザー座標系を設定します。

(2) 使い方：コマンドをクリックして設定画面へ移ると、座標系の名前と座標値を編集します。この座標系は当該プログラムでのみ使用可能です。（使用しない場合、ロボットはデフォルトとして世界座標系を使用します。）

5.1.4.4 ユーザー座標系 () の設定

ユーザー座標系を設定 ワールド ▼

(1) 意味：設定されたユーザー座標系を選択します。

5.1.4.5 負荷 () セントロイド () の設定

荷重設定: 0 kg 重心: X 0 mm Y 0 mm Z 0 mm

(1) 意味：現在の負荷質量とセントロイド位置を設定します。

5.1.4.6 衝突検出度 () の設定

衝突感度の設定 オフ ▼

(1) 意味：現在の衝突検出度を設定します。

5.1.4.7 () 秒待受け設定

🕒 待つ 1

(1) 意味：プログラムがここで () 秒間一時停止します。

注：設定された待受け時間が 0s の場合、ロボットはずっと待ち続けます。

5.1.4.8 () まで待ち続ける

🕒 待つまで

(1) 意味：ある条件が成立するまで待ち続けてから、プログラムを継続実行します。

(2) 使い方：条件フレーム内に判断条件を追加することができます。

5.1.4.9 () 回ループ



(1) 意味：フレーム内のプログラムを () 回ループさせます。

(2) 使い方：() 回のループ実行を希望したいコマンドをこのフレーム内に追加し、プログラムを実行させると、フレーム内のコマンドが () 回までループされます。

5.1.4.10 無限ループ



(1) 意味：そのフレーム内にあるプログラムを無限にループさせます。

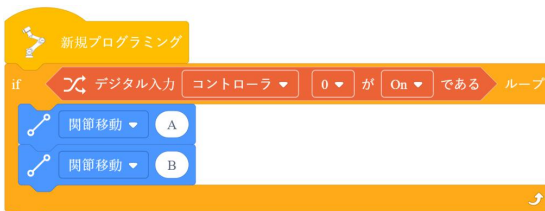
(2) 使い方：ループ実行したいコマンドをこのフレーム内に追加し、プログラムを実行させると、フレーム内のコマンドが無限にループされます。

5.1.4.11 () であれば、ループ実行する



(1) 意味：もしとある条件が成立すると、直ちにフレーム内のプログラムをループさせます。

(2) 例：もしとある条件が成立すると、ロボットが A と B の 2 点の間にループ運動します。



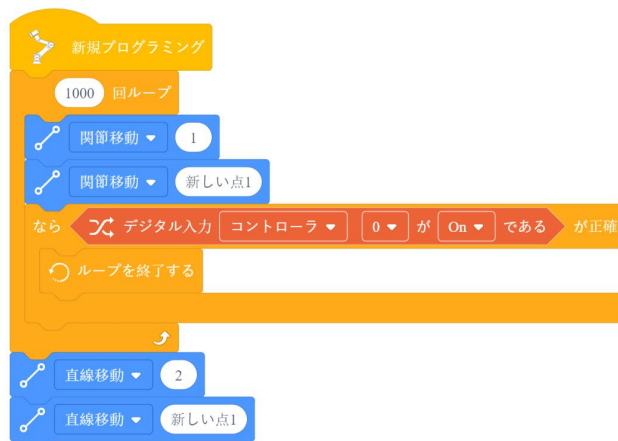
5.1.4.12 終了ループ



(1) 意味：このコマンドのループ状態を終了させます。

(2) 使い方：このコマンドの前に判断条件を追加し、この条件が成立した場合、直ちにループを終了します。

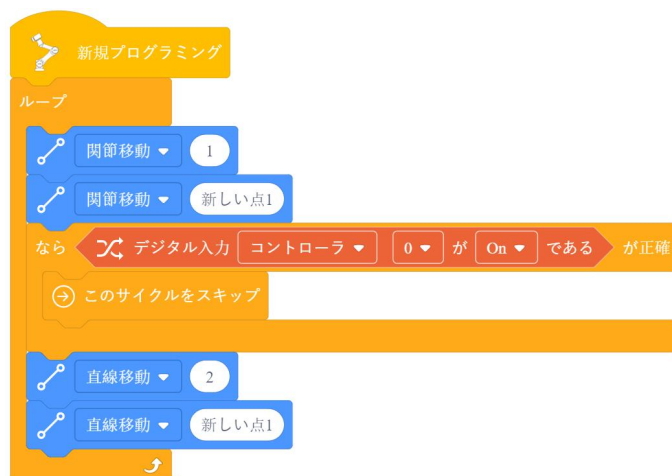
(3) 例：もし DI1 が開の場合、プログラムは直ちにループを終了し、直接「直線運動<2>」から「実行開始」します。



5.1.4.13 今回のループをスキップする



- (1) 意味：今回のループをスキップします。
- (2) 使い方：このコマンドの前に判断条件を追加し、条件が成立した場合、直ちに今回のループをスキップします。
- (3) 例：もし DI1 が開の場合、プログラムは次なる「直線運動<2>」を実行せず、直接ループの始まりである「関節運動<1>」に戻ってから実行します。



5.1.4.14 もし () が真実の場合



- (1) 意味：もしとある条件が成立した場合、フレーム内のプログラムを実行しはじめます。
- (2) 例：もしとある条件が成立した場合に実行します。



5.1.4.15 もし () が真実なら、それ以外の場合は…



(1) 意味：もしとある条件が成立した場合、フレーム内プログラムの前半のみを実行しますが、それ以外の場合は後半プログラムを実行します。

(2) 例：もし a が b より大きい場合、ロボットは A ポイントまで運動しますが、それ以外の場合、ロボットは B ポイントまで運動します。



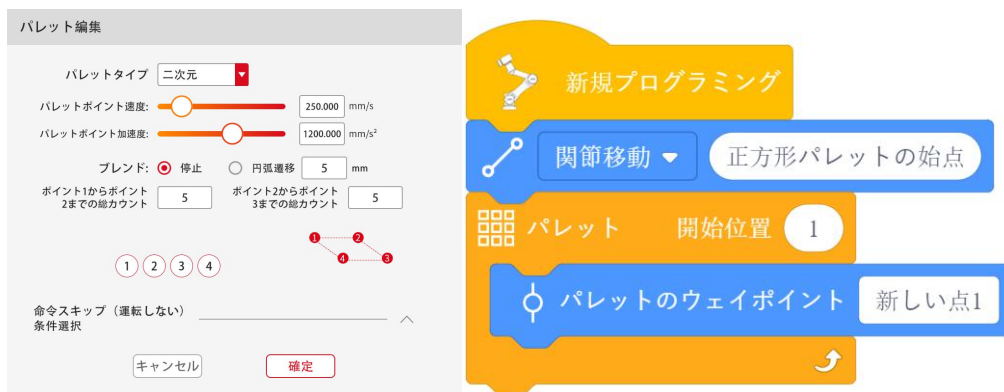
5.1.4.16 パレットの開始位置 ()



(1) 意味：パレットの種類、ポイント位置と間隔数を決めたうえでロボットにパレットプログラムを実行させます。

(2) 使い方：コマンドをクリックすると表示される設定画面では、パレットの種類、速度、加速度を編集すること、ここで停止するか円弧遷移するかを選択すること、パレットポイント間の間隔数を選択すること、パレットのティーチングポイントを編集すること、停止条件を選択すること（プログラムは、ロボットが停止条件で設定された状態を満たすと、このコマンドをスキップします）（開始位置は変数を使用できますが、数値が 0 より小さい場合、プログラムはエラーを報告します。一方、数値がパレット間隔数を超えた場合、パレットの初ポイントから実行します。）ができます。

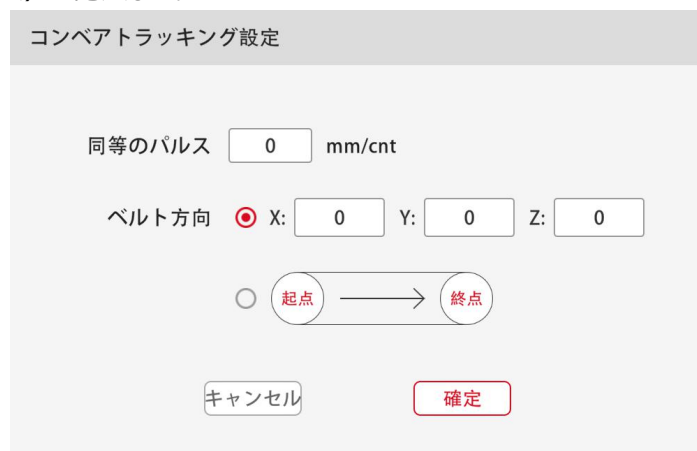
(3) 例：ロボット端末は 3X3 の正方形パレットが配置されます。



5.1.4.17 コンベアトラッキング



- (1) 意味：ロボット端末をコンベアに連動して移動させ、リアルタイムな動きを取らせます。
- (2) 使い方：コマンドをクリックして設定画面へ移り、コンベアの的パルス等量（CNT/mm）と移動方向（ティーチングか直接記入）を記入します。



5.1.4.18 コマンド折畳み



- (1) 意味：コマンドにドロップインしたコマンドを一本のコマンドに折畳み、コマンドが占める空間を少なくします。
- (2) 使い方：折畳みコマンドの名前を編集したうえ、折畳みたいコマンドをドロップインしてコマンドをクリックすると、コマンド折畳みをトリガーできます。
- (3) 例：下記コマンドを折り畳みます。



5.1.4.19 ログタイプ () ログ内容 ()



- (1) 意味：ユーザーがログ内容をカスタマイズして出力します。
- (2) 使い方：コマンドのドロップダウンリストから情報、警告、エラーという三つのログタイプを選び、ログ内容に入力値か文字列変数をドロップインすることができます。また、出力されたログ内容は APP のログ情報にて検索できます。
- (3) 例：指定位置に到達すると、ログ内容 pos1、pos2 を出力します。



5.1.4.20 プログラムの停止或いは一次停止



- (1) 意味：プログラムをとある位置までに停止か一次停止させるように指定します。
- (2) 使い方：コマンドドロップダウンリストでは、一次停止か停止を選ぶことができ、プログラムの任何位置でもこのコマンドを追加することができます。

5.1.4.21 プログラム () の実行



- (1) 意味：ロボット内に格納された他のプログラムを実行します。
- (2) 使い方：コマンドドロップダウンリストでは、プログラミングリスト中の他のプログラムファイルを選ぶことができます。
- (3) 例：例えば、名前が 1 となるプログラムファイルを実行させる例は、下図の通り；



5.1.5 計算コマンド

5.1.5.1 演算子 ()



(1) 意味：6種類の演算方法に対応できます：加算 (+)、減算 (-)、乗算 (*)、除算 (/)、剰余を求める (%)、べき乗計算 (**)。

5.1.5.2 数学関数計算 ()



(1) 意味：サイン、コサイン、タンジェント、逆サイン、逆コサイン、逆タンジェント、自然指数、自然対数、切り上げ、切り下げ、切り上げ、絶対値、平方根を計算する数学関数を提供します。

5.1.5.3 () 演算子 () の比較



(1) 意味：比較用演算子を提供します：より小さい、等しい、より大きい、等しくない、小さいか等しい、大きい等しい。

5.1.5.4 () および ()



(1) 意味：両方の条件が共に成立します。

5.1.5.5 () 若しくは ()



(1) 意味：両方の条件が共に成立するか、片方だけが成立します。

5.1.5.6 () または ()



(1) 意味：両方の条件中に片方だけが成立します。

5.1.5.7 非 ()



(1) 意味：何れの条件も成立しません。

5.1.5.8 位置計算 () 足し算 (減算/逆変換/位置変換) ()

位置計算

加算 ▼

① 位置計算 () 足し算 ()

(1) 意味：2つの位置と姿勢の累計を計算します。

(2) 使い方：位置変数がティーチングポイントをドロップインすることができ、コマンドリターン値が計算結果となります。

② 位置計算 () 引き算 ()

(1) 意味：2つの位置と姿勢の減算を計算します。

(2) 使い方：位置変数がティーチングポイントをドロップインすることができ、コマンドリターン値が計算結果となります。

③ 位置計算の逆変換 ()

(1) 意味：デカルト空間位置の逆変換を計算します。

(2) 使い方：位置変数がティーチングポイントをドロップインすることができ、コマンドリターン値が計算結果となります。

④ 位置計算 () 位置変換 ()

(1) 意味：ロボットの位置と姿勢の変換を計算します。

(2) 使い方：位置変数がティーチングポイントをドロップインすることができ、それぞれがロボット開始姿勢とロボット相対開始位置と姿勢の増分変換となります。コマンド戻り値が計算結果となります。

5.1.5.9 位置距離 () () の計算

位置距離を計算します

(1) 意味：2つの位置と姿勢の間の距離を計算します。

(2) 使い方：位置変数がティーチングポイントをドロップインすることができ、コマンドリターン値が計算結果となります。

5.1.5.10 補間点 () () 係数 ()

補間点

係数: 0

(1) 意味：既定の係数に基づいて2ポイント間の補間点を計算し、係数の数値範囲は[0, 1]とします。

(2) 使い方：位置変数がティーチングポイントをドロップインすることができ、コマンドリターン値が計算結果となります。

5.1.5.11 変換平面 XY (YZ/ZX) 基点 () Z (X/Y) 軸回転 () X (Y/Z) 軸平行移動 () Y (Z/X) 軸平行移動。

変換面

XY ▼

基点

Z軸回転

0

X軸を平行移動

0

Y軸を平行移動

0

(1) 意味：XY (YZ/ZX) 平面内に変換し、基点にてまず Z (X/Y) 軸中心に回転し、そして X (Y/Z) 軸に沿って平行移動し、Y (Z/X) 軸に沿って平行移動します。

(2) 使い方：位置変数がティーチングポイントをドロップインすることができ、コマンドリターン値が計算結果となります。

5.1.5.12 キネマティクスの逆解 (正解)

運動学

逆解 ▼

デカルト値:

基準角度値:

① キネマティクス逆解デカルト空間値

- (1) 意味：選定した参考角度値に基づいて、入力されたデカルト空間位置が同一解空間内にある対応する関節空間位置を計算します。
- (2) 位置変数がティーチングポイントをドロップインすることができ、コマンドリターン値が計算結果となります。

② キネマティクス正解関節角度値

- (1) 意味：入力された関節角度の所在デカルト空間位置を計算します。
- (2) 使い方：位置変数がティーチングポイントをドロップインすることができ、コマンドリターン値が計算結果となります。

5.1.6 文字コマンド

注：文字列コマンドにおいて、対応可能なエスケープ文字は¥¥、¥'、¥"、¥n、¥t、¥r があり、それぞれバックslash記号、シングルクォート、ダブルクォート、改行、水平タブ、キャリッジリターンに対応します。

5.1.6.1 文字列のスティッチング () ()

文字列の結合

- (1) 意味：一定手順に基づいて、ドロップインされた変数や入力された文字列をスティッチングします。
- (2) 使い方：このコマンドをクリックして、ステッチする文字列の数（1～8）と任意の配列区切り記号を編集し、変数をドラッグするか、文字列を入力すると、ステッチされた文字列が戻されます（戻り値は文字列変数タイプ）。

文字列スティッチング編集

数量

行列区切り記号

キャンセル

確定

(3) 例：

例えば、下記変数 A = 8888、B = "Hello"、C =[1、2、3、4、5、6]をステッチングした結果として、以下が戻されました：

A:8888

B:Hello

C:1,2,3,4,5,6

例示プログラムは以下の通り：



5.1.6.2 文字列や配列長さ ()

文字列や配列の長さ

(1) 意味：文字列か配列の長さを計算し、長さ値を戻します。

(2) 使い方：文字列変数、配列変数をドロップインするか文字列を入力することができ、戻り値は長さとなります。

(3) 例：

上位コンピュータが文字列変数「JAKA_AMAZING」の長さ 12 を取得する必要がある場合、配列変数「1,2,3,4,5,6」の長さ 6 を取得する必要がある場合、
例示プログラムは以下の通り：



5.1.6.3 文字列の比較 () ()

文字列の比較

(1) 意味：ASCII コードに基づいて二つ文字列のサイズを比較します。

(2) 使い方：str1、str2 文字列変数のドロップインか文字列値の入力ができます、もし <str1=str2 の場合、0 が戻されます；もし str1str2 の場合、マイナス値が戻されます；もし str1>str2 の場合、プラス値が戻されます。

(3) 例：

例えば、文字列変数 A と a のサイズを比較する場合、戻り値がマイナス値になります。例示プログラムは以下の通り



5.1.6.4 フォーマット出力文字列 フォーマット： () ()

出力文字列の書式設定 書式：

(1) 意味：指定されたデータを、指定されたフォーマットの文字列として出力します。

(2) 使い方：このコマンドをクリックして、フォーマットするデータ数（1～8）と任意の配列区切り記号を編集し、変数または入力値をドラッグすることができます。また、このコマンドはフォーマットされた文字列を返し、戻り値は文字列変数タイプになります。

文字列フォーマット編集

数量

2

行列区切り記号

キャンセル

確定

(3) 例：

例えば、ロボットの関連情報を取得したい場合。

Current joint position: [0,90,0,90,180,0]

End load: [0,0,0,0]

例示プログラムは以下の通り：



5.1.6.5 フォーマット文字列の入力 文字列： () フォーマット： () () 結果

書式設定文字列を入力 文字列: 書式: 結果

(1) 意味：特定のフォーマットの文字列をマッチし、マッチングデータを指定変数に入力して、結果を返します。

(2) 使い方：このコマンドをクリックすると、フォーマットするデータ数（1～8）と任意の配列区切り記号を編集し、指定した変数または入力値をドラッグし、フォーマットが成功した場合は一致した変数の数を、そうでない場合は-1を返します。フォーマットされた結果変数は、整数型、浮動小数点型、文字列型、または定数をサポートしています。

文字列フォーマット編集

数量

2

行列区切り記号

キャンセル

確定

(3) 例：

文字列変数「End load is : 0、0、0」中の配列[0、0、0]を抽出する場合、例示プログラムとして如下のものがあります：



5.1.6.6 フォーマット文字列の入力 文字列：() フォーマット () ()

書式設定文字列を入力 文字列: 書式:

(1) 意味：特定フォーマットの文字列をマッチし、そのマッチングデータを指定変数の中に入力します。

(2) 使い方：このコマンドをクリックすると、フォーマットするデータ数（1～8）と任意の配列区切り記号を編集し、指定した変数または入力値をドラッグし、指定変数は必要な数値に抽出することができます。フォーマットされた結果変数は、整数型、浮動小数点型、文字列型、または定数をサポートしています。

文字列フォーマット編集

数量

行列区切り記号

(3) 例：

文字列変数「End load is : 0、0、0」中の配列[0、0、0]を抽出する場合、例示プログラムとして如下のものがあります：



5.1.6.7 配列を文字列に転換 配列：() 区切り記号：()

配列を文字列に変換 配列: 区切り記号:

(1) 意味：配列を特定フォーマットの文字列変数に変換します。

(2) 使い方：「配列」の後には転換したい配列種類の変数をドロップインし、区切り記号には転換したい配列区切り記号を入力することができます。

(3) 例：

例えば、配列変数[1、2、3、4、5、6]を「1a2a3a4a5a6」に転換する場合の示例プログラムは以下のものがあります：



5.1.6.8 文字列を配列に転換 文字列： () 区切り記号： ()

文字列を配列に変換 文字列: () 区切り記号: ()

(1) 意味：特定フォーマットの文字列を配列変数に転換します。

(2) 使い方：「文字列」の後には配列種類に転換したい文字列変数をドロップインし、区切り記号には展開したい配列区切り記号を入力することができます。

(3) 例：

例えば、文字列変数「1a2a3a4a5a6」を配列変数[1、2、3、4、5、6]に転換したい場合の例示



5.1.6.9 配列： () 下付け文字へのアクセス： ()

配列: () 取得インデックス: 0

(1) 意味：配列変数中の要素をアクセスします。

(2) 使い方：アクセスしたい配列変数をドロップインするかアクセスしたい配列下付け文字値を入力することができます。下付け文字インデックスは 0 の順からスタートします。コマンド戻り値がアクセスした要素となります。

(3) 例：

例えば、配列変数[1、2、3、4、5、6]中に下付け文字が 3 の値をアクセスする、即ち戻り値が 4 の場合の示例プログラムは以下の通り：



5.1.6.10 配列： () 下付け開始： () 下付け終了： () ステップ値： ()

配列: () 開始インデックス: 0 終了インデックス: 0 ステップ値: 1

(1) 意味：必要な下付け文字とステップ値に基づき、配列変数中の要素をアクセスします。

(2) 使い方：アクセスする配列変数をドラッグして、アクセスする配列の下付け文字とステップ値を入力すると、下付けインデックスが 0 から始まるようになっています。コマンド戻り値がアクセスするサブ配列となっています。

(3) 例：

例えば、配列変数[1、2、3、4、5、6]中の下付け文字が 0、2、4 となる値をアクセスする場合、即ち戻り値が[1、3、5]となる示例プログラムは以下となります：



5.1.6.11 配列要素の設定 名前： () 下付け文字： () 値： ()

配列要素の設定 変数: インデックス: 値:

(1) 意味 : 配列中のとある要素の値を設定します。

(2) 使い方 : 設定する配列変数をドラッグして、設定する配列の下付け文字を入力すると、下付けインデックスが 0 から始まるようになっています。

(3) 例 :

例えば、配列変数[1、2、3、4、5、6]中の下付け文字が 0 を 888 と設定する場合、即ち配列変数が[888、2、3、4、5、6]となる示例プログラムは以下となります :



5.1.7 通信コマンド

5.1.7.1 SOCKET () IP アドレスをオンにします () ポート : ()



(1) 意味 : TCP Client を新規作成し、TCP server との通信を構築します。

(2) 使い方 : ドロップダウンリストにて指定する SOCKET ID を選び、TCP Server の IP アドレスとポート番号を入力すると、このコマンドを実行する際に、TCP Client (ロボット) と TCP Server との接続が構築されます。

5.1.7.2 SOCKET () IP アドレスの有効化 : () ポート : () 結果



(1) 意味 : TCP Client を新規作成し、TCP server との通信を構築してから、Socket ハンドルを返します。

(2) 使い方 : ドロップダウンリストで指定された SOCKET ID を選択し、TCP server の IP アドレスとポート番号を入力してから、このコマンドを実行すると、TCP クライアント (ロボット) と TCP サーバーの間で接続が確立されます。このコマンドは、接続が成功した場合は 0 以上の整数値を返し、接続が失敗した場合は -1 を返します。

5.1.7.3 SOCKET () の無効化



(1) 意味 : 指定した SOCKET の通信接続を切ります。

(2) 使い方 : ドロップダウンリストで指定された SOCKET ID を選択し、このコマンドを実行すると、指定された SOCKET の通信接続が切れます。

5.1.7.4 SOCKET () 受信変数 : () 待ち時間 : () s



(1) 意味 : () s 以内変数を受信完了するまで待受ける。コントローラがデータクエリ文字列を送り、そして受信待受け状態に入ります。(対応しない配列タイプ)

(2) 使い方：ドロップダウンリストで指定した SOCKET ID を選択し、受信する変数の種類をドラッグして入力し、()秒以内に受信した場合、または変数を受信せずに待ち時間が()秒を超えた場合はプログラムを続行します（待ち時間を 0 秒に設定した場合は、条件が満たされるまでロボットは待機します）、Unicode 文字を含む変数名はサポートされていません。

注：受信変数の種類はデジタル型、文字列型と配列型があり、サーバーデータの発信フォーマットは：

デジタル型：<変数名前><データ内容>

文字列型：<変数名前><"データ内容">

配列型：<変数名前><[データ内容]>

5.1.7.5 SOCKET () 発信：()



(1) 意味：コントローラに、SOCKET 通信経由して変数を TCP Server に送らせます。

(2) 使い方：ドロップダウンリストで指定した SOCKET ID を選択し、変数をドラッグインするか、値を入力してこのコマンドを実行すると、TCP サーバはこの変数の内容を受信します。

注：サポートしている変数の種類は、デジタル型、文字列型と配列型であり、発信データのフォーマットに特別な要求はないが、Unicode 文字には対応しません。

5.1.7.6 SOCKET () 発信：() 結果



(1) 意味：コントローラに、SOCKET 通信経由して変数を TCP Server に送らせます。

(2) 使い方：ドロップダウンリストで指定された SOCKET ID を選択し、変数ドラッグするか値入力してから、このコマンドを実行すると、TCP Server がこの変数の内容を受信できるようになります。このコマンドは、接続が成功した場合は 0 以上の整数値を返し、接続が失敗した場合は-1 を返します。

5.1.7.7 SOCKET () 受信配列 配列長さ：() 待ち時間：() s



(1) 意味：長さ()の配列変数を受信するまで、ロボットは()秒間走行を停止し、コントローラはデータクエリ文字列を送信した後、受信待ち状態になります。

(2) 使い方：ドロップダウンリストで指定された SOCKET ID を選択し、変数が()秒以内に受信された場合、または受信されずに待ち時間が()秒を超えた場合は、プログラムを続行します（待ち時間が 0 秒に設定されている場合は、ロボットは条件が満たされるまで待ちます）。受信に失敗した場合、または受信がタイムアウトした場合は、空の配列が返され、受信が成功した場合は、すべてが配列に存在していることになります。

注：サーバーデータ発信フォーマット：[n1,n2,n3...]

5.1.7.8 SOCKET () 受信待受け時間：() s



(1) 意味：ロボットが () s 内に、データが受信されるまで運動を停止します。

(2) 使い方：ドロップダウンリストで指定された SOCKET ID を選択し、データが()秒以内に受信された場合、または受信されずに待ち時間が()秒を超えた場合はプログラムを続行し（待ち時間を 0 秒に設定した場合は、条件が満たされるまでロボットは待機します）ます。正常に受信された場合、関数によって受信した文字列が返されます。受信に失敗するかタイムアウトした場合、空の文字列を返します。

注：変数受信タイプは文字列タイプで、サーバー側のデータ送信フォーマットに特別な要求はありませんが、Unicode 文字はサポートされていません。

5.1.7.9 リフレッシュ信号量 識別子 () レート ()

 信号量を更新する 識別名 なし ▼ 周波数 0

(1) 意味：「ハードウェアと通信」設定端末 IO において定義された信号量をリフレッシュします。

(2) 使い方：コマンドドロップダウンリストにて、リフレッシュしたい信号量を選び、リフレッシュレートを単位 Hz で入力します。リフレッシュレート：すべての信号のリフレッシュレートの合計は、バスの広帯域性によって制限される 125Hz を超えないように、システム内部で自動的に調達されています。

5.1.7.10 信号量状態の取得 識別子 ()

 信号量状態を取得する 識別名 なし ▼

(1) 意味：信号量の値を取得します。

(2) 使い方：コマンドドロップダウンリストにて必要な信号量を選び、コマンド戻り値は当該信号量の現在値となります。

5.1.7.11 Modbus コマンドの発送 チャンネル () バイト配列 ()

 Modbusコマンド送信 チャンネル TIO RS485H ▼ 文字配列 ☐

(1) 意味：TIO 外部装置に対する即時制御コマンドの発信。

(2) 使い方：コマンドドロップダウンリストにて使用する TIO チャンネルを選び、ユーザーはバイト配列に 16 進数のデータコマンドを入力できますが、ユーザーによる校正コードの追加は不要で、システムが自動的に追加できます。

(3) 例：

とある装置の位置を 500 に指定し、設定済み位置情報を取得する例とするプログラムは以下の通り：



5.1.8 サブプログラムコマンド

5.1.8.1 サブプログラム

サブルーチン:  新規プログラミング

① コマンド類のサブプログラム

(1) 意味：一連のコマンドシーケンスを比較的独立したプログラムセグメントとして組合せ、プログラミングにこのセグメントを追加すると、該当のコマンドシーケンスが実行されるほか、実行が完了すると元の位置に戻ってプログラムを続行するようになっています。この相対的独立したプログラムセグメントこそサブプログラムとなります。

(2) 使い方：コマンド編集タイプのサブプログラムを一つ追加し、サブプログラムにて単独なプログラムセグメントを編集して保存すると、このサブプログラムをメインプログラムの中でいつでも追加し、重複利用することができます。

② スクリプト類のサブプログラム

(1) 意味：サブプログラムの中身は、所定のフォーマットで書かれたスクリプト構文です。

(2) 使い方：スクリプト編集類のサブプログラムを追加し、このサブプログラムにて必要な所定フォーマットのスクリプト構文を編集して保存すると、いつでもメインプログラムにこのサブプログラムを追加して再利用することができます。

5.1.8.2 サブプログラムの追加か削除

① サブプログラムの追加

サブプログラム列のタイトルの右にあるプラスボタンをクリックし、追加するサブプログラムの種類を選択し、確認をクリックすると追加が完了します。 作成完了した後に、プログラミングエリアが自動的にサブプログラムへとジャンプします。

② サブプログラムの削除

サブプログラム列のタイトルの右にある「マイナス」ボタンをクリックし、削除したいサブプログラム右側のゴミ箱アイコンをクリックして確定します。

5.1.8.3 サブプログラムの編集

サブプログラムコマンド列かプログラムにて、編集したいサブプログラムコマンドボックスをクリックすると、サブプログラム編集画面に移ることができます。サブプログラムの編集が終わったら、メインプログラムを再度保存して、サブプログラムがメインプログラムで正しく変更されていることを確認します。

5.1.9 変数コマンド

5.1.9.1 変数の追加

変数欄のタイトルの右にあるプラスボタンをクリックし、追加したい変数の種類を選択し、確定をクリックすると、変数設定画面に入り、必要なパラメータを入力して確定すると、変数の追加が完了します。変数追加は、システム変数、プログラム変数、スピード変数と位置変数の4種に対応できます。

① システム変数

システム変数は、設定画面のシステム変数にて作成（詳しくは章節 3.1.4.3）する、或いはプログラム制御変数コマンド欄にて「+」をクリックして追加するという二つの方法から選ぶことができます。システム変数はコントローラに格納されているもので、ロボットの電源を切ったり、プログラムを削除したりしてもクリアされたり失われたりすることはなく、任意プログラムによって呼び出すことができます。システム変数のタイプはデジタル型で、変数名前は中国語形式に対応します。

変数を追加

☒ システム変数 ☐ プログラム変数 ☐ 速度変数 ☐ 位置変数

変数タイプ: 数字

変数名: 初期値:

キャンセル

確定

② プログラム変数

プログラム変数は、現在開いているプログラムで使用するために作成された変数です。これらの変数は、現在のプログラムに追従し、現在開いているプログラムでのみ呼び出して使用することができ、他のプログラムでは呼び出して使用できません。プログラムを削除すると、このプログラムで作成したプログラム変数も削除されます。プログラム変数は、デジタル類、文字列類と配列類の三種類が含まれます。

注：この変数作成画面では、変数の初期値を作成するためのフォーマットを記入する必要はありません。

中国語の変数名にも対応していますが、第三者との通信では、エンコードの問題で中国語形式の変数名によって文字化けとなることがありますので、注意が必要です。

変数を追加

☐ システム変数 ☒ プログラム変数 ☐ 速度変数 ☐ 位置変数

変数タイプ: 数字

変数名: 初期値:

キャンセル

確定

③ スピード変数

スピード変数は特殊な変数として、プログラムにおいてコマンド中のグローバルスピードコマンドを移動する方法によって呼び出し使用したり、クローズすることができます。実際の稼働環境に合わせて、ロボットの速度をグローバルに制御することができます。グローバル速度コマンドで速度変数を呼び出すと、このコマンド以降の動作速度は、現在呼び出されている速度変数で選択されている速度に追従し、運動コマンドでは速度設定は反映されません。

注：スピード変数は、関節速度、関節加速度、直線速度、直線加速度などの内容として socket コマンドで送信することができます。

変数を追加

☐ システム変数 ☐ プログラム変数 ☒ 速度変数 ☐ 位置変数

変数名:

☒ 関節速度: 60.000 °/s
 関節加速度: 200.000 °/s²

☒ 直線速度: 250.000 mm/s
 直線加速度: 250.000 mm/s²

キャンセル

確定

④ 位置変数

位置変数は特定な変数として運動コマンドのみに適用可能で、他のコマンドによって呼び出すことができません。位置変数は、ロボットのデカルト位置と関節の位置を設定することができます。テキストボックスをクリックして直接入力するか、「編集」をクリックしてロボットの位置決めのためのマニュアル操作インターフェースに入ることができます。ただし、デカルト位置を設定する際には、ユーザー座標系とツール座標系によく注意する必要があり、設定した座標系と実際のロボットポイントの位置が一致しないと、ズレなどの安全上の問題が発生します。位置変数を運動コマンドに取り込む際に、ロボットは運動タイプに応じた変数中のポイント情報を選択しながら動作します。

変数を追加

☐ システム変数 ☐ プログラム変数 ☐ 速度変数 ☒ 位置変数

変数名:

デカルトの位置:

X	Y	Z	RX	RY	RZ
0	0	0	0	0	0

関節位置:

関節一	関節二	関節三	関節四	関節五	関節六
0	0	0	0	0	0

ユーザー座標系:

ツール座標系:

キャンセル

編集

確定

5.1.9.2 削除変数

変数列のタイトルの右にあるマイナスボタンをクリックし、削除したい変数の右にあるゴミ箱アイコンをクリックして確定します。

5.1.9.3 変数パラメーターの編集

変数コマンド欄、或いはプログラムにて編集したい変数ボックスをクリックすると、変数編集画面へ移ります。変数編集が完了した後に、それが正しくメインプログラムにて反映されるように、いま一度メインプログラムを保存し直す必要があります。

5.1.9.4 変数 () を () に等しく設定します

設置 システム変数:count ▼ = 0

(1) 意味：変数に値を割り当てる。

(2) 使い方：入力ボックスには一定値、変数、SOCKET から取得した値等を入力し、そのデータを変数に割り当てます。

注：

配列変数の全体に値を割り当てる場合、具体的な数値の前後に[]を付け加える必要があります。

設置 プログラム変数:array ▼ = [0,0,0,0,0]

文字列に対して値を割り当てる場合、具体的な内容の前後に""（英文字のコロン）を付け加える必要があります。

設置 プログラム変数:string ▼ = "JAKA"

5.1.10 拡張コマンド

5.1.10.1 一定カスミージングパラメーターの設定

 恒久力柔順パラメータ設定。

(1) 意味：設定一定力モード下のスミージングパラメーター。

(2) 使い方：力制御感知機能を有効にする方向の選択、各方向の減衰力の設定（プログラム実行時の外部環境の剛性が高いほど、減衰力を大きく設定する必要がある。ドラッグモードでは 10N 以上、M は 0.2Nm 以上を推奨、0 には設定できない）、反発力（ロボットがあらかじめ設定された軌道に戻るためのもので、設定値が大きいほどロボットはあらかじめ設定された軌道から外れにくい）、一定力の大きさ（ロボットの端末と外部環境との接触力が、設定された一定力の範囲内であることを保証する）、方向トラッキング機能（ノーマルトラッキング）を設定することができます。

恒久力柔順制御編集

	方向	減衰力	跳ね返す	恒常力	速度	方向トレース
☐	Fx	0.000 N	0.000 Nm	0.000 N	1m/s	☐
☐	Fy	0.000 N	0.000 Nm	0.000 N	1m/s	☐
☐	Fz	0.000 N	0.000 Nm	0.000 N	1m/s	☐
☐	Mx	0.000 Nm	0.000 Nm	0.000 Nm	90°/s	☐
☐	My	0.000 Nm	0.000 Nm	0.000 Nm	90°/s	☐
☐	Mz	0.000 Nm	0.000 Nm	0.000 Nm	90°/s	☐

5.1.10.2 一定カスミージング制御をオンにします 初期化/初期化しない

 恒常力柔順制御をオンにする。 初期化 ▼

(1) 意味：一定力モードを有効にし、初期化するか否かを選択します。

(2) 使い方：初期化：センサーのオフセットや負荷などの補正（このモードに入るときは、センサーに外部接触力がないようにします）を行うが、初期化はしません。直近の補正値を採用します。

5.1.10.3 一定カスミージング制御を閉じる

 恒久力柔順制御をオフにします。

(1) 意味：一定力モードを終了します。

5.1.10.4 スピードスミージングパラメーターの設定

 速度柔軟パラメータ設定

(1) 意味：設定スピードモード下のスミージングパラメーター。

(2) 使い方：減速のレベリングを最高三段階に設定することができます。ロボット端末の荷重が制御力設定値を上回った場合、ロボットはセンサーの検出値が制御力設定値以内に入るまで、減速モードを続けます。

速度柔順制御編集

縮小モード	速度レベル (%)
☐ 1	
☐ 2	
☐ 3	

キャンセル 確定

5.1.10.5 スピードスミージング制御をオンにします 初期化/初期化しない

 スピード柔順制御をオンにする 初期化 ▼

(1) 意味：スピードモードをオンにして、初期化するかどうかを選択します。

(2) 使い方：初期化：センサーのオフセットや負荷などの補正（このモードに入るときは、センサーに外部接触力がないようにします）を行うが、初期化はしません。直近の補正値を採用します。

5.1.10.6 スピードスミージング制御を閉じる

 スピードソフトをオフにします。

(1) 意味：スピードモードを終了します。

5.1.10.7 スミージング制御力

 柔軟制御力

- (1) 意味：制御力の方向と強弱を設定します。
- (2) 使い方：スピードスムージング制御モードにおける制御力を設定するためのものです。

柔軟制御力編集

方向	制御力	
Fx	0.000	N
Fy	0.000	N
Fz	0.000	N
Fy	0.000	N
Fz	0.000	N
Fx	0.000	N

キャンセル
確定

5.1.10.8 運動終了条件



- (1) 意味：運動に対する設定条件を事前に終了させます。
- (2) 使い方：監視したい方向を確認し、上限値または下限値を設定し、接触外力値が下限値未満または上限値以上になると、運動終了条件がトリガーされます。このコマンドは、直後の運動コマンドを監視し、運動終了条件が発生した場合、ロボットは現在の位置から次の運動コマンドの設定位置まで即座に移動します。ただし、「一定荷重スムージングパラメータ設定」ですべてのパラメータが0に設定されていることと、方向がチェックされていないことが必須条件です。

運動終了条件

	方向	下限	下 限	上限	上 限
☐	Fx	☐	0.000 N	☐	0.000 N
☐	Fy	☐	0.000 N	☐	0.000 N
☐	Fz	☐	0.000 N	☐	0.000 N
☐	Mx	☐	0.000 Nm	☐	0.000 Nm
☐	My	☐	0.000 Nm	☐	0.000 Nm
☐	Mz	☐	0.000 Nm	☐	0.000 Nm

キャンセル
確定

5.1.10.9 力制御座標系 ツール座標系/世界座標系の設定



- (1) 意味：力制御モードに使われる座標系（座標系方向とセンサー方向が一致していること）を設定します。
- (2) 使い方：ツール座標系を選び、ロボット端末位置と姿勢がツール座標系にて記述されています。世

界座標系を選び、ロボット端末位置と姿勢がロボットベース座標系にて記述されています。

5.2 画面見直し

5.2.1 拡大

画面右下にあるプラスボタンをクリックし⁺、プログラミングエリアの画面を拡大します。

5.2.2 縮小

画面右下にあるマイナスボタンをクリックし⁻、プログラミングエリアの画面を縮小します。

5.2.3 元に戻す

画面右下にある復元ボタンをクリックし[≡]、プログラミングエリア画面をデフォルト寸法に戻します。

5.3 プログラム操作

5.3.1 プログラム実行

実行をクリックすると  実行、ロボットが現在アップロードされたプログラムを実行し始めます。

プログラム実行中に、実行ボタンが一次停止プログラム  暂停 と終了プログラム  停止 を実行させるための一次停止ボタンと停止ボタンへと変わります。

5.3.2 プログラム実行速度

スピードをクリックすると  速度、スピードボタンの下にスピードバーが表示されます。スピードバーをスライドさせることで、ロボットの現在実行スピードを調節すること、或いはスピードバーの下にある数字をクリックすることで、直接実行スピードのパーセントを入力することができます。

5.3.3 プログラムの新規作成

プログラム追加  ボタンをクリックしてプログラムを作成し、プログラム上方の名前をクリックすると、そのプログラム名前を修正することができます。

5.3.4 プログラムを保存する

保存  ボタンをクリックすると、現在のプログラムを最新バージョンとして保存します。

5.3.5 プログラム名を付けて保存する

「名前を付けて保存  する」をクリックし、新しいファイル名を入力してから確定すると、現在のプログラムを別名保存することができます。

5.3.6 プログラムを開く

プログラミングプロジェクトリストをクリックすると 、制御盤に保存されているすべてのプログラムがプロジェクトリストに表示され、名前、作成時間、ファイルサイズの 3 つの方法でソートすることができます。

5.3.6.1 プログラムを開く

- (1) 意味：保存されたプログラムを開いてアップロードします。
- (2) 使い方：開きたいプログラムの名前をクリックすると、当プログラムがアップロードされます。

5.3.6.2 プログラムのインポート

(1) 意味：装置において、APP からエクスポートした JAKA ロボットのプログラムを現在接続中のコントローラにインポートします。

(2) 使い方：プログラミング・プロジェクト・リストウィンドウの右上にある「インポートボタン」をクリックし 、プログラムの圧縮ファイルが保存されているデバイス内のパスを選択し、そのパスの下にインポートするプログラムの圧縮ファイルを見つけて選択し、確認をクリックするとインポートが完了します。

5.3.6.3 プログラムのエクスポート

- (1) 意味：コントローラに保存されているプログラムを現在装置の中にエクスポートします。

(2) 使い方：プログラミング・プロジェクト・リストウィンドウの右上にある「エクスポート」ボタンをクリックし 、エクスポートするプログラムを選択して「確認」をクリックし、エクスポートするファイルのパス・アドレスを選択して「確認」をクリ

ックすると、エクスポートが完了します。同時に最高 5 つまでのジョブプログラムのエクスポートが可能です。

5.3.6.4 プログラムの削除

(1) 意味：ロボットプログラムを削除します。

(2) 使い方：プログラミング・プロジェクト・リストウィンドウ右上の「削除」から 、削除するプログラムをクリックして確定します。同時に最高 5 つまでのジョブプログラム削除が可能です。

5.3.6.5 プログラムの共有化

(1) 意味：ネットワークを介して、現在コントローラ中のプログラムを同じサブネットにある他の JAKA Zu ロボットコントローラにコピーしていきます。

(2) 使い方：プログラミング・プロジェクト・リストウィンドウの右上にある「共有化」から 、共有化するプログラムをクリックして確定し、ダイアログウィンドウにてロボットを選んで確定します。

5.3.7 プログラムの高度操作

高度操作をクリックすると、プログラム編集エリアの上に「全選択」、「コピー」、「削除」、「キャンセル」ボタンが表示されます。

5.3.7.1 全選択

(1) 意味：現在編集エリアにある全てのコマンドを選択します。

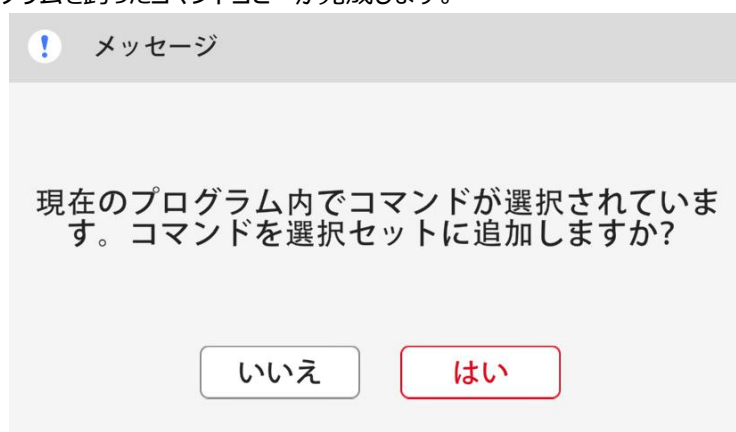
(2) 使い方：「選択」をクリックする

5.3.7.2 コピー

(1) 意味：セレクトした順に、選定されたコマンドをコピー、現在編集エリアの中に貼り付けます。

(2) 使い方：コピーするコマンドを任意の順序で選択し、「コピー」をクリックします。

注：プログラムをまたいでコピーする場合は、コピーしたいコマンドを確認した後、貼り付けたいプログラムやサブプログラムを開くと、図のようなダイアログが表示されますので、確定をクリックしてください。プログラムに入ってから、コピー集合をクリックすると、プログラムを跨ったコマンドコピーが完成します。



5.3.7.3 削除

(1) 意味：選定したコマンドを削除します。

(2) 使い方：削除したいコマンドを選んで、削除をクリックします。

5.3.7.4 キャンセル

(1) 意味：高度操作を終了します。

(2) 使い方：キャンセル をクリックします。

5.3.8 プログラムのシングルステップデバッグ

(1) 意味：プログラムコマンドのシングルステップを実行します。



(2) 使い方：シングルステップデバッグボタンをクリックすると、ロボットがプログラムの実行を開始します。コマンドの左に赤い矢印が表示され、現在どのコマンドが実行されているかを示します。「次へ」をクリックすると、ロボットが次のコマンドを実行しながら、ユーザーによる再びの「次へ」操作を待受けします。デバッグ中に、「デバッグ中止」や「停止」をクリックすると、プログラム実行が終了されます。

5.3.9 プログラムロック



プログラムロックを有効にすると、プログラムが修正されませんが、再度修改を再開したい場合、まずプログラムロックを解除する必要があります。

5.4 プログラミングの仕方

5.4.1 プログラム的実行ロジック

プログラムの名前が表示されているコマンドボックスがプログラムのヘッド部で、ヘッドと一緒に並んでいるすべてのコマンドがプログラム主体を構成しています。ロボットがプログラムを実行する際に、プログラムのヘッド部から上から下への順に実行します。

5.4.2 コマンドの使い方

JAKA ロボットのプログラムがグラフィカルに作成されるため、ユーザーはコマンドバーから必要なコマンドを直接ドラッグ&ドロップし、各コマンドのパラメータを編集してプログラムを作成することができます。

その他、ユーザーはコマンドボックスの形に基づき、当該コマンドの働きとプログラムにおける位置を初期判断できるため、コマンドの相互ロジック関係への理解に有利となります。

5.4.2.1 動作コマンド

「関節運動または直線運動」や「もし () が真実なら」のような形状のコマンドボックスは、運動コマンドと呼ばれます。この種のコマンドまで実行した場合、ロボットやコントローラはそれ相応な動作を行います。

動作コマンドは互いにシリアル接続によって、一個の完全なるプログラムにすることもできます。



5.4.2.2 判定コマンド

「比較コマンド」や「デジタル入力 () () は (ON/OFF)」のような形状のコマンドボックスを判定コマンドと呼びます。判定コマンドの役割は判定条件として、「もし () が真実なら」、「もし () が真実なら、さもなければ」、「() まで待受ける」等コマンドの判定ボックス内に格納されています。



5.4.2.3 データコマンド

「アナログ入力 () () の取得」、演算子コマンド、変数などの形状と類似したコマンドボックスで、データコマンドと呼びます。

データコマンドは、通常データの取得か保存用として、データコマンドの形と似たデータボックスに格納されています。



付録

付録一 ロボットパラメーターのデータ種類

1 現在関節位置の取得：

データタイプ：配列

長さ：6

意味：配列中の六つ要素は、それぞれ一関節から六関節までの角度値（単位°）を表しています。

2 現在ツール側のセンター位置の取得：

データタイプ：配列

長さ：6

意味：配列の6つの要素は、現在のユーザ座標系における、現在使用しているツール座標系の原点の空間的位置を、0番目の要素から5番目の要素まで表しており、順にX、Y、Z（単位：mm）、RX、RY、RZ（単位：°）に対応します。

3 現在フランジセンターの位置と姿勢の取得：

データタイプ：配列

長さ：6

意味：ロボット端末フランジのセンターが現在ユーザ座標系における空間位置として、0番目から5番目の要素を表し、順にX、Y、Z（単位mm）、RX、RY、RZ（単位°）に対応します。

4 現在端末負荷の取得：

データタイプ：配列

長さ：4

意味：現在保存されているロボットの端末荷重に関する情報は、0番目から3番目の要素まで、順に荷重の質量（kg）と、フランジの中心に対する荷重の質量中心のX、Y、Zの距離（mm）に対応します。

5 現在端末力の取得：

データタイプ：配列

長さ：6

意味：現在の端末トルクセンサーが端末の負荷を補正した後の正味のトルク値（単位N.m.）を取得します。

6 現在感度の取得：

データタイプ：ナンバー

意味：現在ロボットに設定されている衝突感度を設定します。

7. システム時刻：

データタイプ：ナンバー

意味：現在システム時刻の取得。

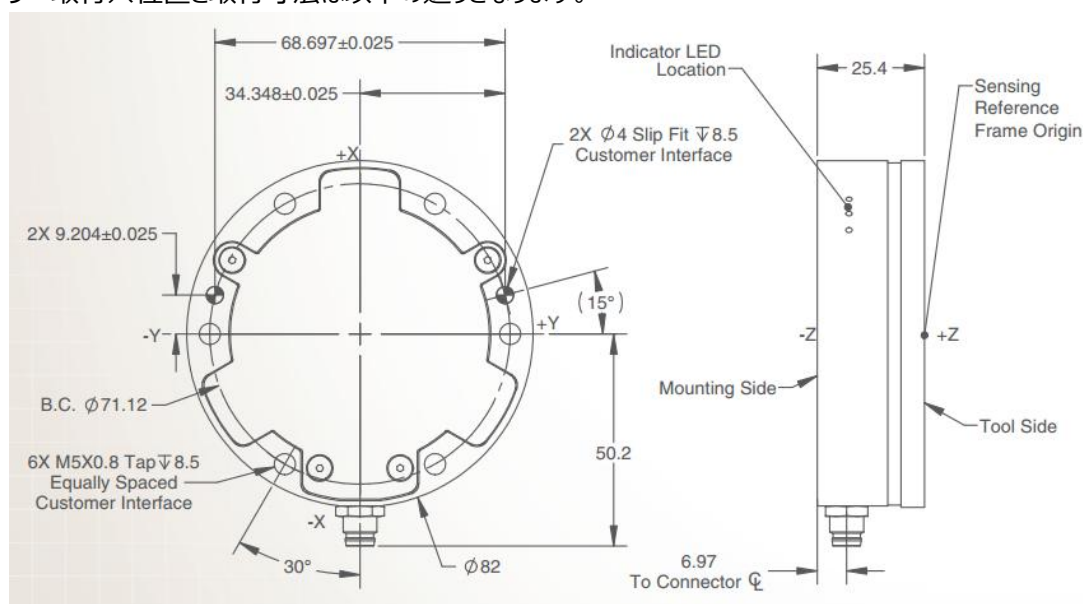
付録二 力制御センサー I 型

1 概要

本マルチ型六軸力/トルクセンサーは、同時に 3 つの力と 3 つのトルク測定に対応できます。このセンサーは、加えられた力による「ツール側フランジ」と「本体」との間の相対的な変形を検出し、センサーの弾性単位の変化量を抵抗線ひずみ計で測定します。このセンサーには、抵抗線ひずみ計からの信号の変化をリアルタイムに収集・処理する内蔵式システムが搭載されており、加えられた力の大きさと方向を高精度・高応答でリアルタイムに出力します。センサーの使用に当たっては、出力効果が影響されないように正しく取付けてください。

2 センサーの取付

センサー取付穴位置と取付寸法は以下の通りとなります。

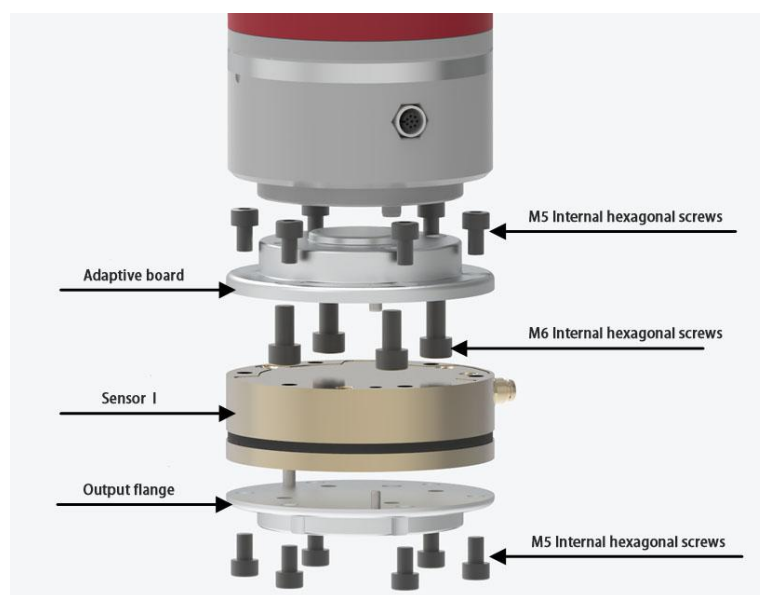


センサー寸法

a. 取り付け前に、取り付け具、アダプタープレート、センサーの表面に損傷や異物がないことを確認してください。異物などにより接触が不均一な場合、取り付ける機器（またはアダプター）の間に隙間ができ、製品の IP64 性能が保証できません、実際のセンサーの出力効果にも影響を与えます。

b. センサーをアダプタープレートから分離し、アダプタープレートをロボットの端末フランジに取り付けます。ただし、センサーの座標系の XY 方向は、センサーを設置する前にあらかじめマウントしておくことで、ロボットの端末フランジ座標系の XY 方向と同じになることが保証されるように注意してください。つまり、ロボットの端末フランジの中心が示す TIO 方向は、センサーの-Y 方向と一致にさせます。取付方向が一致しないと、その後の使用に影響します。

c. センサーとアダプタープレートを固く密着して取り付けます。6 本の M5 六角穴付きネジを使って、センサーがアダプタープレートに均等に接触するように、ネジを対角線沿いに締めていきます。



センサーと装置の取付イメージ図

d. 出力フランジとセンサーの出力側と繋ぎ合わせます。出力フランジのメカニカルインターフェースとロボット端末フランジのメカニカルインターフェースと一致させます。

3 センサーの基礎パラメーター

Fx/Fy(N)	200	Fz(N)	360
Mx/My(Nm)	8	Mz(Nm)	8
過負荷水準(%)	500	精確さ(%)	0.5
精度(%)	0.1	保護レベル	IP64
稼働温度(℃)	5~80	給電電圧(V)	12~24
通信インターフェース	Ethernet	システム分解能(Bit)	16

精度：精度とは、複数の測定値の一致度、すなわち、入力を同じ方向に複数回テストしたときにセンサーが得る出力チャートの一貫性の度合いを示す指標です。繰り返し精度とは、出力誤差の標準偏差を公称出力に対する割合（%FS）で表したものです。

精確さ：精確さとは、測定値が真の値からどの程度乖離しているかを示す指標です。精確さは、出力量と理論上の真の値との偏差値を公称出力に対する割合（%FS）で表したものです。

4 取扱注意事項

- a 仕様外の温度・湿度では使用しないでください。
- b 配線が完全に正しいこと。電源を入れる際には、接続ケーブルの色が本マニュアルに記載されている形と一致していることを確認してください。接続端子に誤りがあると、センサーの内部回路がショートして破損する恐れがありますので、必ず注意して確認してください。
- c センサーには、内蔵式システムなどの精密パーツが組み込まれており、関連する振動・衝撃試験を実施していますが、製品の落下や過度の振動は故障の原因となりますのでご注意ください。
- d センサーの取り付け時には、強く叩いたりしないでください。特にアダプタープレートとの嵌合において、アダプタープレートの加工などにより鉢合わせのギャップが厳しい場合でも、センサーの性能を損なう恐れがありますので、センサーを叩いたりしないでください。
- e センサーを設置して電源を入れた後、動作させる前に 1 時間ほどウォーミングアップすることをお勧めします。
- f 実際に使用する際には、センサーに搭載する機器の質量を考慮して、過負荷にならないようにする必要があります。
- g 使用中に疑問や不具合が生じた場合は、無断で操作しないで、当社に直接お問い合わせください。

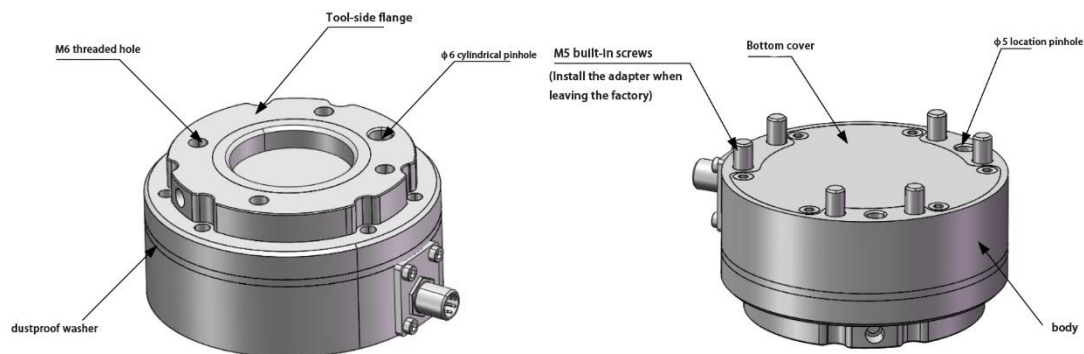
5 センサー取付ボルトの締付トルクを目安値

メートル法	締付トルク参考値 (Nm)
M2	0.4
M3	2.0
M4	4.0
M5	8.0
M6	13.0
M8	35.0

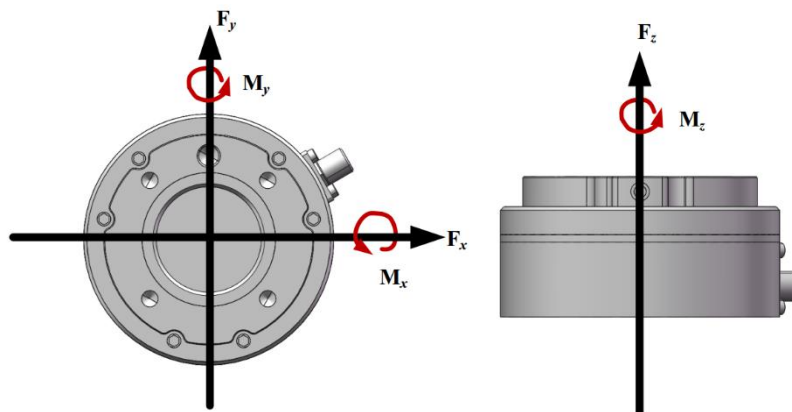
付録三 力制御センサー II 型

1 概要

本マルチ型六軸力/トルクセンサーは、同時に 3 つの力と 3 つのトルク測定に対応できます。このセンサーは、加えられた力による「ツール側フランジ」と「本体」との間の相対的な変形を検出し、センサーの弾性単位の変化量を抵抗線ひずみ計で測定します。このセンサーには、抵抗線ひずみ計からの信号の変化をリアルタイムに収集・処理する内蔵式システムが搭載されており、加えられた力の大きさと方向を高精度・高応答でリアルタイムに出力します。センサーの使用に当たっては、出力効果が影響されないように正しく取付けてください。

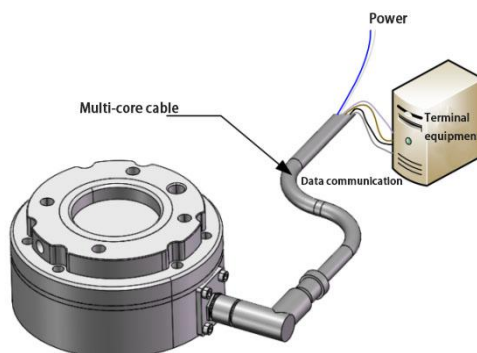


センサー外観



センサー座標系の定義

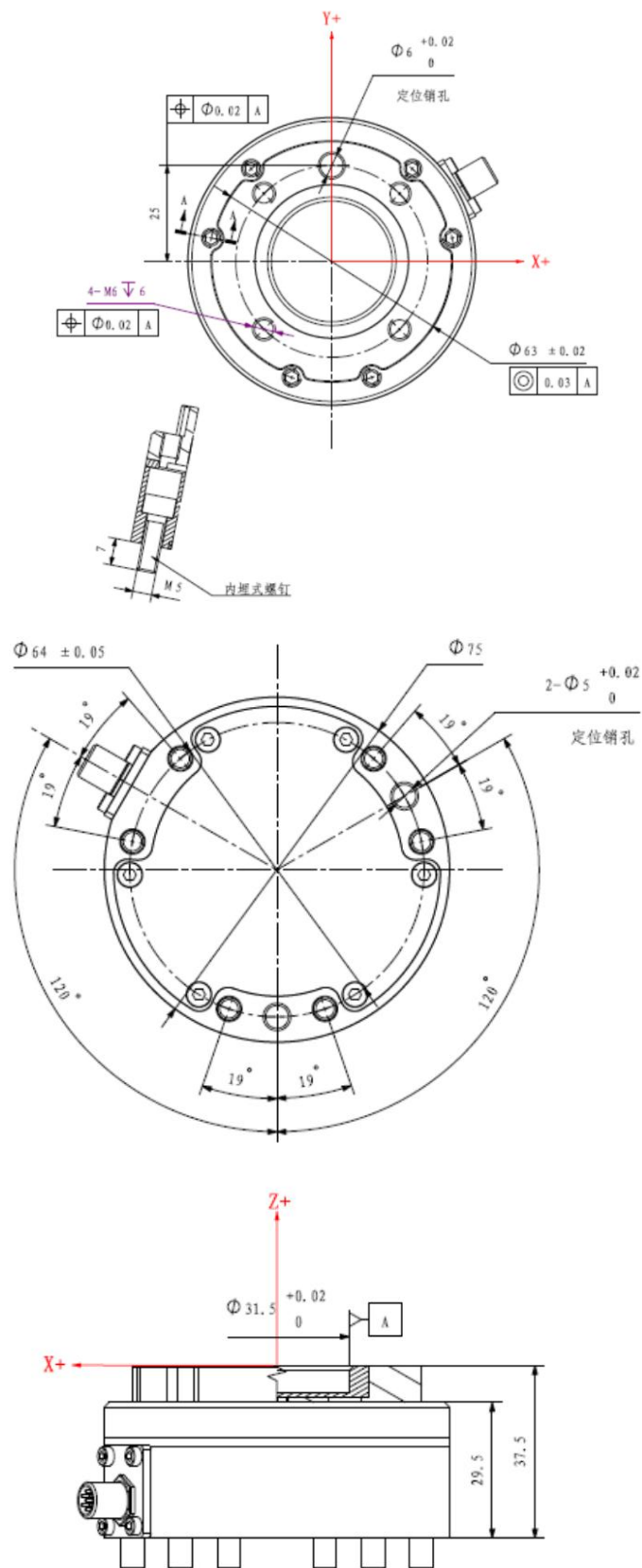
ひずみゲージからの電圧信号は、センサー本体に内蔵式システムが組み込まれによってリアルタイムに処理され、実際の荷重値へと変換されてからデジタル信号の形で出力されます。



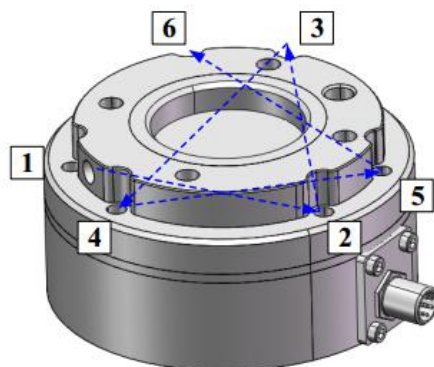
センサーの電気接続イメージ図

2 センサーの取付

センサー取付穴位置と取付寸法は以下の通りとなります。



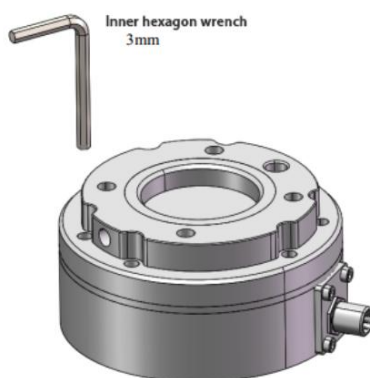
ネジを締めるときは、図の対角線上の順序で徐々に締めていきます。センサーと装着する機器やアダプターの工具を均一に接触させるようにします。



ネジの締付手順

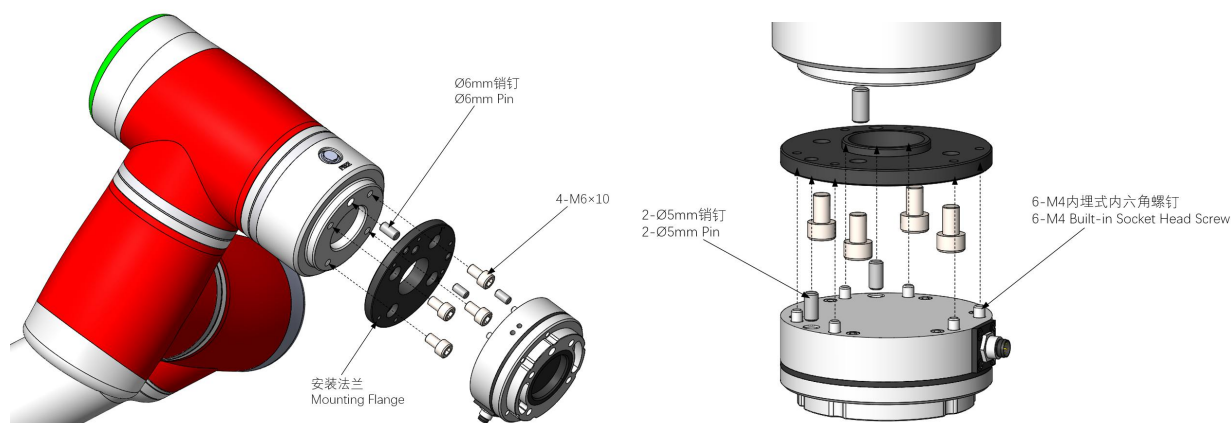
a. 取り付け前に、取り付け具、アダプタープレート、センサーの表面に損傷や異物がないことを確認してください。異物などにより接触が不均一な場合、取り付ける機器（またはアダプター）の間に隙間ができ、製品の IP64 性能が保証できません、実際のセンサーの出力効果にも影響を与えます。

b. センサーとアダプタープレートを分離し、アダプタープレートを取り付けるデバイスに取り付けます。本センサーの出荷状態は、アダプタープレートをセンサー本体に 6 本の埋め込み式六角ネジで取り付けた形となります。3mm の六角レンチを使って 6 本の埋め込み式六角ネジを緩め、アダプタープレートをセンサー本体から分離します。アームエンドアダプタープレートは、φ6 の位置決めピンで設置する機器との位置関係を決め、製品パッケージに含まれる M6 標準六角ネジ 2 本と M6 非標準六角ネジ 2 本を使用して、アダプタープレートを設置する機器に固定します。位置決めピンは、機器を設置する際の再現性を取るためのもので、位置決めピンを使用しなくてもセンサーの性能に影響はありません。



3mm の六角レンチで埋込式ネジを緩めます

c. センサーとアダプタープレートを固く密着して取り付けます。位置決めピンでセンサーとアダプタープレートとの位置合わせを行い、センサー取付方向と装置の使用方向が一致することを確認します。センサーの埋め込みネジ（6 個）で固定します。六角レンチ（幅 3mm）をセンサーのツール側フランジの取り付け穴に差し込み、ネジを右方向に回して固定します。ネジの締付手順は下図に従って行います。センサーの保護水準 IP64 を確保するために、センサーを取り付けた後、6 つの M5 プラグネジをツール側フランジにある 6 つの対応するねじ穴にねじ込むことで、センサーの内部チャンバーを外部環境からある程度隔離しておくことができます。



センサーと装置の取付イメージ図

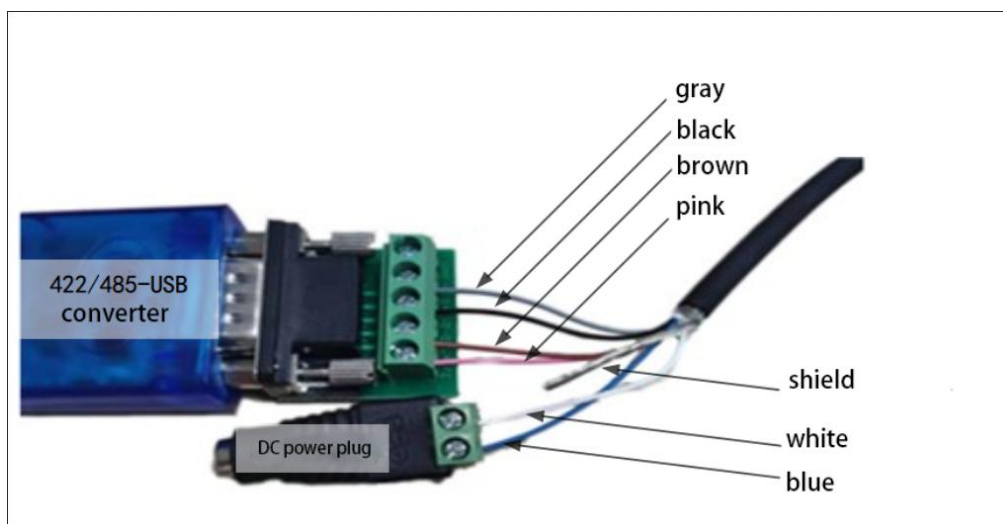
d. 装置のツールインターフェースとセンサーツール側フランジと接続します。センサーツール側フランジには、装置ツールの接続用として、4つのM6ネジ穴及びφ6ピン孔となる汎用型インターフェースが配属されています。センサーツール側のフランジ位置決めピンは、機器ツールの再現性を取るためのもので、これがなくてもセンサーの性能に影響を与えません。

e. 接続ケーブルは製品に同梱されています。接続ケーブルは、センサーのケーブルインターフェースに合わせた形の多芯ケーブルを使用しています。下図のように、多芯ケーブルのコネクターをセンサーのケーブルインターフェースに合わせて、押し込みます。IP64の性能を得るためには、押し込んだ後に、多芯ケーブルのインターフェースにあるネジ接続ハウジングが固く締付けられ、ケーブルが緩まないようにしておくことが重要です。設置の際の個々のケーブルの芯線の配線は、与えられた芯の色の定義に厳密に従い、電源を逆方向に接続するとセンサーが破損する可能性があるため、注意が必要です。



ケーブル接続操作

番号	芯線の色	定義
1	青	電源+
2	白	電源-
3	ピンク	422 バス線センサー受信+
4	ブラウン	422 バス線センサー受信-
5	黒	422 か 485 バス線センサー発信+
6	グレー	422 か 485 バス線センサー発信-
7	シールド	



422/485-USB コンバータと電源プラグとの接続イメージ図

3 センサーの基礎パラメーター

Fx/Fy(N)	200	Fz(N)	200
Mx/My(Nm)	8	Mz(Nm)	8
過負荷水準(%)	300	精確さ(%)	0.5
精度(%)	0.1	保護レベル	IP64
稼働温度(℃)	5~80	給電電圧(V)	9~24
通信インターフェース	RS485	サンプリング解像度 (Bit)	24

精度：精度とは、複数の測定値の一致度、すなわち、入力を同じ方向に複数回テストしたときにセンサーが得る出力チャートの一貫性の度合いを示す指標です。繰り返し精度とは、出力誤差の標準偏差を公称出力に対する割合（%FS）で表したものです。

精確さ：精確さとは、測定値が真の値からどの程度乖離しているかを示す指標です。精確さは、出力量と理論上の真の値との偏差値を公称出力に対する割合（%FS）で表したものです。（%FS）

4 取扱注意事項

- 仕様外の温度・湿度では使用しないでください。
- 配線が完全に正しいこと。電源を入れる際には、接続ケーブルの色が本マニュアルに記載されている形と一致していることを確認してください。接続端子に誤りがあると、センサーの内部回路がショートして破損する恐れがありますので、必ず注意して確認してください。
- センサーには、内蔵式システムなどの精密パーツが組み込まれており、関連する振動・衝撃試験を実施していますが、製品の落下や過度の振動は故障の原因となりますのでご注意ください。
- センサーの取り付け時には、強く叩いたりしないでください。特にアダプタープレートとの嵌合において、アダプタープレートの加工などにより鉢合わせのギャップが厳しい場合でも、センサーの性能を損なう恐れがありますので、センサーを叩いたりしないでください。
- センサーを設置して電源を入れた後、動作させる前に 1 時間ほどウォーミングアップすることをお勧めしま

す。

f 実際に使用する際には、センサーに搭載する機器の質量を考慮して、過負荷にならないようにする必要があります。

g 使用中に疑問や不具合が生じた場合は、無断で操作しないで、当社に直接お問い合わせください。

5 センサー取付ボルト締付ボルトの目安

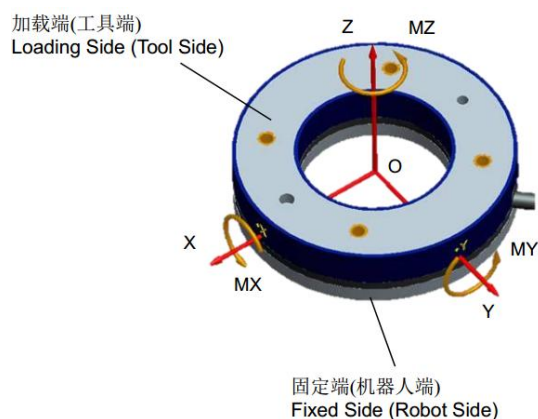
メートル法	締付トルク参考値 (Nm)
M2	0.4
M3	2.0
M4	4.0
M5	8.0
M6	13.0
M8	35.0

付録四 力制御センサー III 型

1 概要

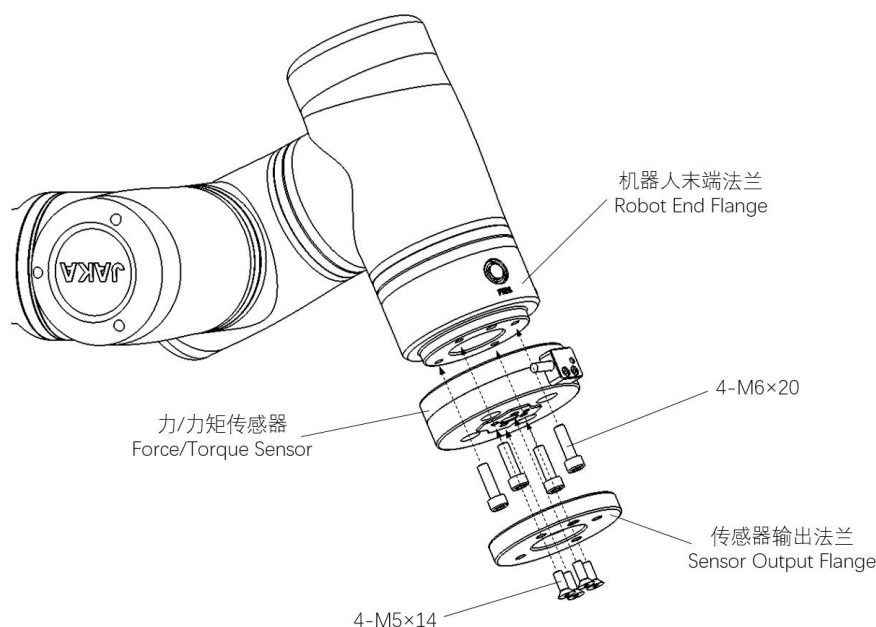
本マルチ型六軸力/トルクセンサーは、同時に 3 つの力と 3 つのトルク測定に対応できます。六軸力制御センサーは、一般的に固定側（ロボット側）と負荷側（ツール側）に分かれています。両側に相対的な力が加えると、センサーが弾性変形を内部の抵抗型ひずみ計によって電圧の信号に変換されてから出力します。

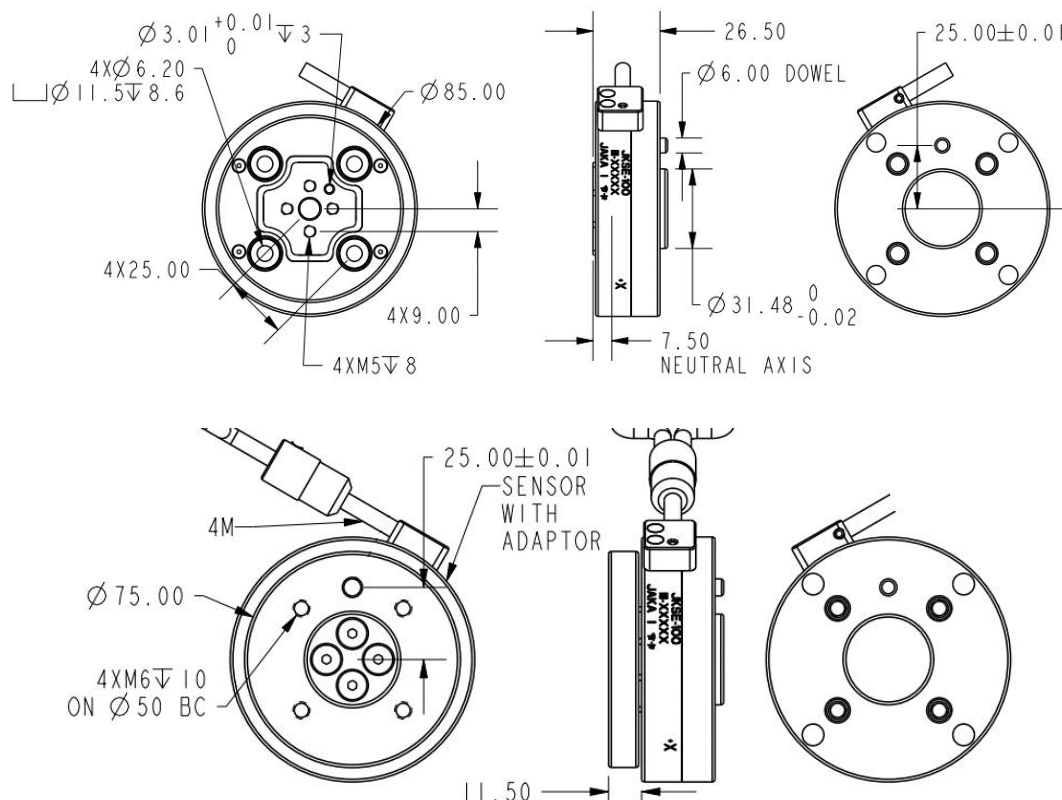
六軸力制御センサーの出力力とモーメントは、中立座標系に対するもので、中立座標系は一般的にセンサーの幾何学的中心に位置しています。



2 センサーの取付

センサー寸法、固定側（ロボット側）位置と負荷側（ツール側）位置、取付方法は下図に示した通り。電線やコネクターは固定側に固定されています。電線の揺れや引っ張りが力制御センサーの測定に影響を与えないように、固定側を固定して、負荷側から外力を加える仕組みにする必要があります。





ネジを締めるときは、対角線上の手順で徐々に締めていきます。センサーと装着する機器やアダプターの工具を均一に接触させるようにします。

a. 取付前は、必ず装置とセンサーの表面に傷や異物がないことを確認してください。異物などにより接触が不均一な場合、取り付ける機器（またはアダプター）の間に隙間ができ、製品の IP64 性能が保証できません、実際のセンサーの出力効果にも影響を与えます。

b. 位置決めピンは、機器を設置する際の再現性を取るためのもので、位置決めピンを使用しなくてもセンサーの性能に影響はありません。

c. 接続ケーブルは製品に同梱されています。接続ケーブルは、センサーのケーブルインターフェースに合わせた形の多芯ケーブルを使用しています。下図の通り、電源は DC 24V で給電され、正極と負極を逆に接続されるとセンサーが破損しますのでご注意ください。センサーのデフォルト IP は 192.168.0.108 で、変更や修正についてはセンサー III 設定の説明書に参照するか、当社に連絡してください。ネットワークポートをロボットコントローラと同一セグメントに接続する。



番号	芯線の色	定義
----	------	----

1	青	電源+
2	白青	電源-
3	黒	シールド線

3 取扱注意事項

- 仕様外の温度・湿度では使用しないでください。
- 配線が完全に正しいこと。電源を入れる際には、接続ケーブルの色が本マニュアルに記載されている形と一致していることを確認してください。接続端子に誤りがあると、センサーの内部回路がショートして破損する恐れがありますので、必ず注意して確認してください。
- センサーには、内蔵式システムなどの精密パーツが組み込まれており、関連する振動・衝撃試験を実施していますが、製品の落下や過度の振動は故障の原因となりますのでご注意ください。
- センサーの取り付け時には、強く叩いたりしないでください。センサーの性能を損なう恐れがありますので、センサーを叩いたりしないでください。
- センサーを設置して電源を入れた後、動作させる前に 1 時間ほどウォーミングアップすることをお勧めします。
- 実際に使用する際には、センサーに搭載する機器の質を考慮して、過負荷にならないようにする必要があります。
- 使用中に疑問や不具合が生じた場合は、無断で操作しないで、当社に直接お問い合わせください。

4 センサーの基礎パラメーター

Fx/Fy(N)	100	Fz(N)	100
Mx/My(Nm)	8	Mz(Nm)	8
過負荷水準 (%)	200	精確さ(%)	0.5
精度(%)	0.5	保護レベル	IP64
稼働温度(℃)	-40~100	給電電圧(V)	24
通信インターフェース	Ethernet	サンプリング解像度(Bit)	24

精度：精度とは、複数の測定値の一致度、すなわち、入力を同じ方向に複数回テストしたときにセンサーが得る出力チャートの一貫性の度合いを示す指標です。繰返し精度とは、出力誤差の標準偏差を公称出力に対する割合（%FS）で表したものです。

精確さ：精確さとは、測定値が真の値からどの程度乖離しているかを示す指標です。精確さは、出力量と理論上の真の値との偏差値を公称出力に対する割合（%FS）で表したものです。（%FS）

5 センサー取付ボルトの締付トルク目安値

メートル法	締付トルク参考値 (Nm)
M2	0.4
M3	2.0
M4	4.0
M5	8.0
M6	13.0
M8	35.0

付録五 ヴィジュアル保護システム

1 背景

産業用ロボットアームは、様々な産業分野で広く利用されています。ロボットアームは、部品の把持や組立、マテリアルハンドリングなどを容易に行うことができるため、人間とコンピューターの協働や人間とコンピューターのインタラクションが重要な開発の方向性となっています。JAKA の助ちゃんシリーズ協働ロボットは、これらの伝統的な作業を持ち運び可能かつ安全に行うことに特化しているだけでなく、ロボットが動作中に人や環境と相互作用にも注目しているため、ロボットの安全ガードがますます不可欠になっています。

2 目的

本製品は、独立したコアプロセッサと画像処理システムを搭載した産業用広角カメラを使用して、ロボットコントローラーとの接続を確立するように設計されています。図 1 に示した通り、JAKA 視覚保護システムは、ロボットの作業領域の上部にカメラを設置して作業現場を監視し、監視領域に外部の物体（人や機器）が侵入すると、取得した画像をもとに処理・判断を行い、人や機械の安全を確保するための適切な処置を行うようロボットにコマンドを出すシステムです。

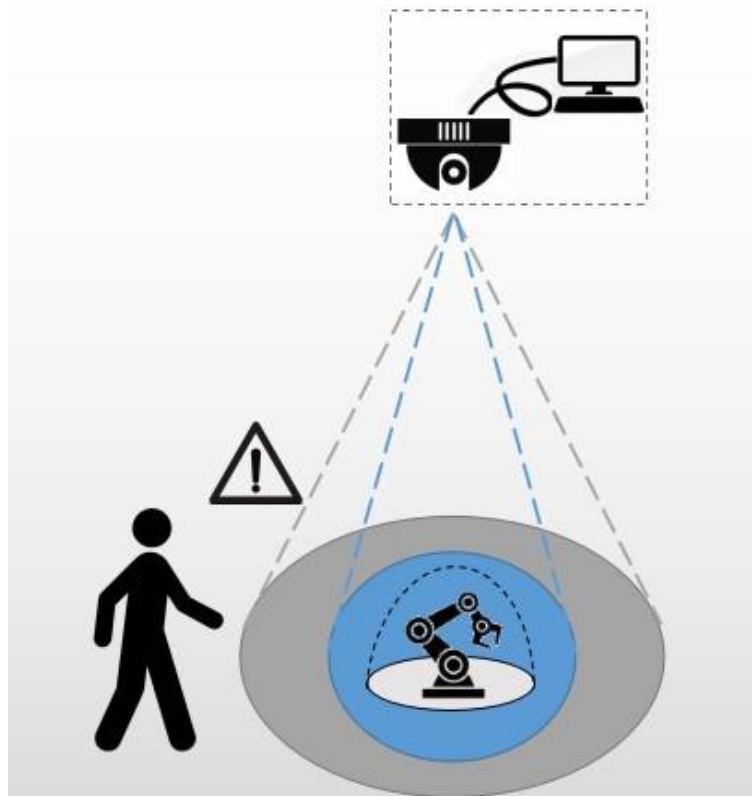


図 1 視覚保護のイメージ図

3 製品及び機能に関する説明

製品概要

視覚保護システムは、ハードウェアとソフトウェアで構成されており、安全保護機能を実現するためには、視覚保護デバイス、APP、ロボット、コントローラの相互作用が必要です。ロボットとコントローラ以外に、図 2 に示す視覚保護装置が本製品の主要ハードウェア装置です。本装置には、プロセッサとカメラが内蔵されています。ソフト上の操作は、図 3 の JAKA 助ちゃん APP における視覚保護モジュールによって実現されます。



図 2 視覚保護装置の現物イメージ



図 3 視覚保護システム APP の操作画面

製品機能と特徴

JAKA 視覚保護システムは以下の機能特徴があります：

- 視覚保護システムには、独立した APP モジュールが搭載されており、APP 上で保護エリアをカスタマイズし、実際に保護する必要のあるエリアを設定することができます。設定完了後、エリア情報がプロセッサに保存され、次回起動する前には、設定されたエリアがそのまま反映されます。
- エリア設定完了後に、システムはレベル 2 の安全保護機能を備える視覚保護状態に入ります。第 1 階層のエリア（減速エリア）として物体が侵入した場合、システムがそれを検知して制御部に結果を送り、ロボットは減速を開始します。第 2 階層のエリア（協働エリア）として物体が侵入した場合、システムがそれを検知して制御部に結果を送り、ロボットは停止するようになっています。
- 本製品には、独立したビデオストリーミングサーバーが搭載されており、ビデオストリーミングアドレスを介して保護領域のライブ映像にアクセスし、APP でリアルタイムに監視することができます。或いは他の端末を利用してリアルタイムのモニタリングも実現できます。

4 システム構築と使用

図 4 に示した通り、JAKA 視覚保護システムは、複雑なハードウェア機器を必要とせず、内蔵されたカメラとコアプロセッサで、安全保護機能を実現することができます。カメラは保護エリアの情報を得るために使用され、ネットワークポートは POE 電源供給と通信の両方に使用でき、HDMI インターフェースは外付け表示スクリーンによるリアルタイムシーンや保護情報の外部表示に使用でき、IO ポートは外部のロボットコントローラや他のハードウェアデバイスが安全保護作業を完了するための IO 制御情報を出力するために使用できます。以上のように、本システムは、カメラ、画像処理モジュール、

IO 制御、TCP 通信を 1 つのシステムに統合しており、このような小型の視覚保護システムだけで、ユーザーがカスタマイズした保護エリアの設定やシステムのセキュリティ保護と安全監視を実現することができます。



図 4 視覚保護装置の説明

図 5 は完全なる JAKA 視覚保護システムの構築仕方を示したものです。保護エリアから一定の高さ（推奨 3m）にシステムを固定しますが、ユーザーの保護エリアの実際の大きさに応じて設置高さを調整することができます。各コンポーネント間の通信を確立するために、すべてのモジュールを同じネットワークセグメント（ルーター）の下に設定しました。視覚保護装置の IP は現在 192.168.2.141 に固定されているため、通信を確立するためにはルーターも 192.168.2.1 に設定する必要があります。視覚保護装置はイーサネット POE 受電によって給電され、ネットワークケーブルを介して POE 電源に接続することができます。

詳しい構築手順は：

- a. ルーターネットワーク情報を設定して給電します。
- b. POE 電源で給電し、ネットワークケーブルを介してルーターとの接続を確立します。
- c. ロボットコントローラもネットワークケーブルでルーターと接続します。
- d. 視覚保護装置は、ロボットの上の高さに取り付けブラケットによって取り付けられており、その高さは監視領域の範囲によって決定されます。取付高さが 3m の場合、視覚保護装置の視野はおおよそ 5m×4m の長方形のエリアになります。

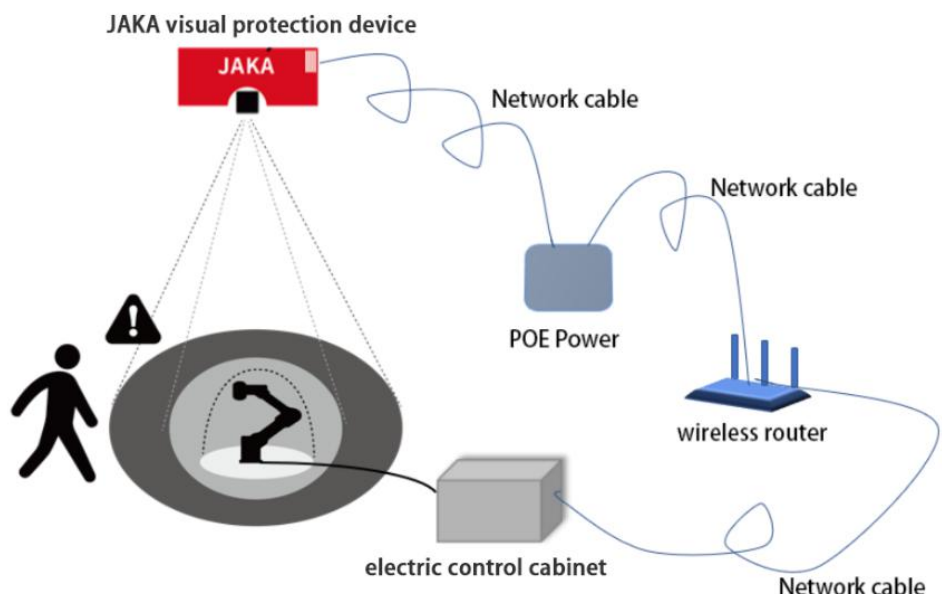


図 5 JAKA 視覚保護システムの稼働構造のイメージ図

安全エリアの設定及び保護機能を起動して

ネットワークケーブルで視覚保護装置をPOE電源に接続すると、システムが動作し始めます。すべてのプログラムがシステム上で起動するように設定されているので、視覚保護システムのオン/オフの切り替えは、ネットワークケーブルの抜き差しで行います。視覚保護システムをオンにすると、APP上で保護エリアを設定することができるが、ルーターの対応する無線ネットワークに接続する必要があります。

● APPによる拡張IOの設定

以下の手順でAPPに拡張IOを設定してください。拡張IOが設定完了してオンにすると、コントローラは視覚保護装置との通信を確立し、視覚保護装置の監視情報を取得することができます。拡張IOの設定手順は以下の通りで、IPアドレスとポート番号は図に従ってください。

拡張IOを追加した後は、DI0を縮小モードに設置して、DI1を保護停止モードに設定してください。これにて拡張IO機能の設定は完了です。視覚保護がオンになっていることを確認した後、拡張IOを実行に設定すれば、視覚保護デバイスとコントローラ間のリアルタイム通信が確立されます。



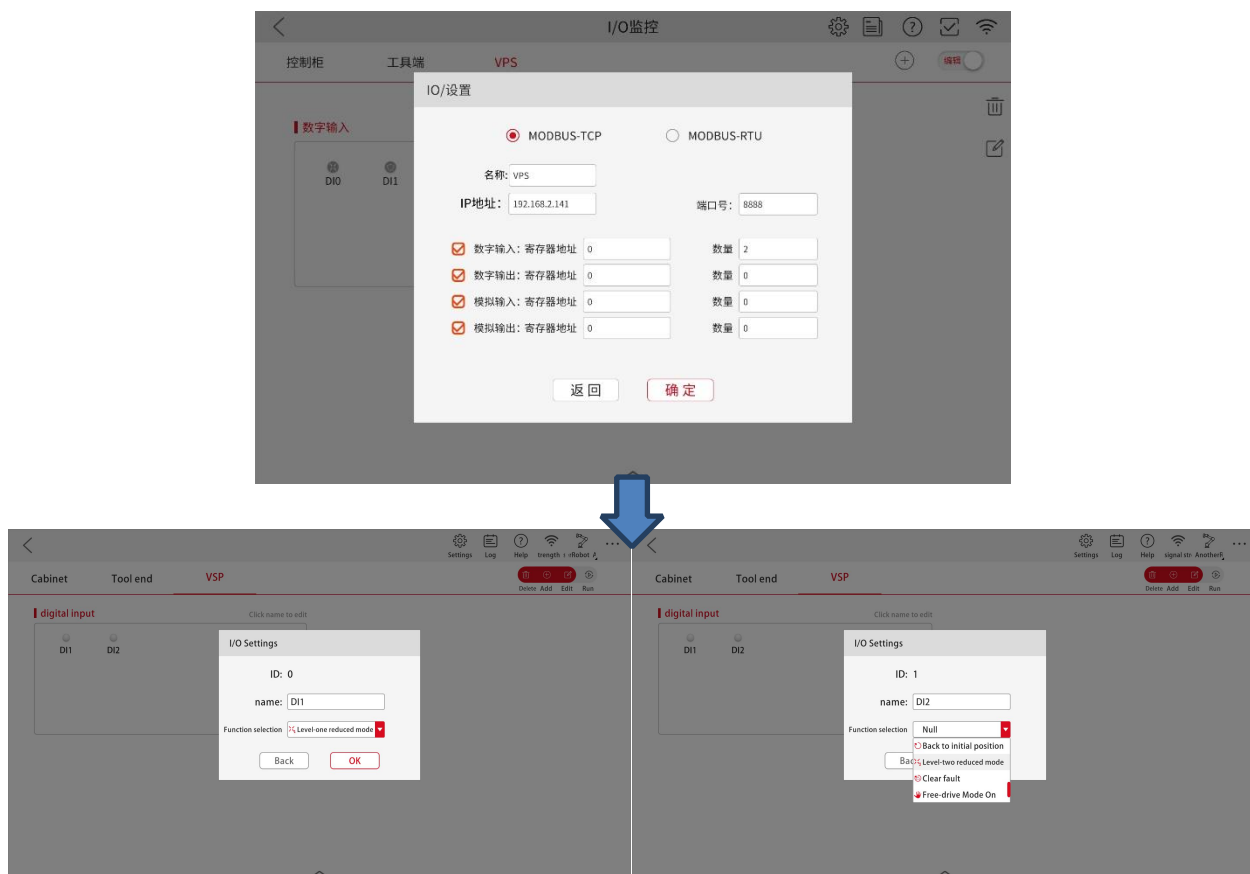


図 6 APP の拡張 IO 設定

● APP による保護エリアのカスタマイズ設置

本製品は、APP 端末を介して保護エリアをカスタマイズできるほか、エリア設定に幅広い形状にも対応しているため、自主選択ができます。

図 7 に示した通り、APP で視覚保護モジュールを開き、ボタンをクリックして接続を設定します。図 8 に示した通り、設定したい視覚保護装置の IP と保護領域の目標形状を記入して設定画面にアクセスします。本品は長方形、円形及びカスタムカーブという三種類の保護エリア形状に対応できます。



図 7 APP の視覚保護モジュール

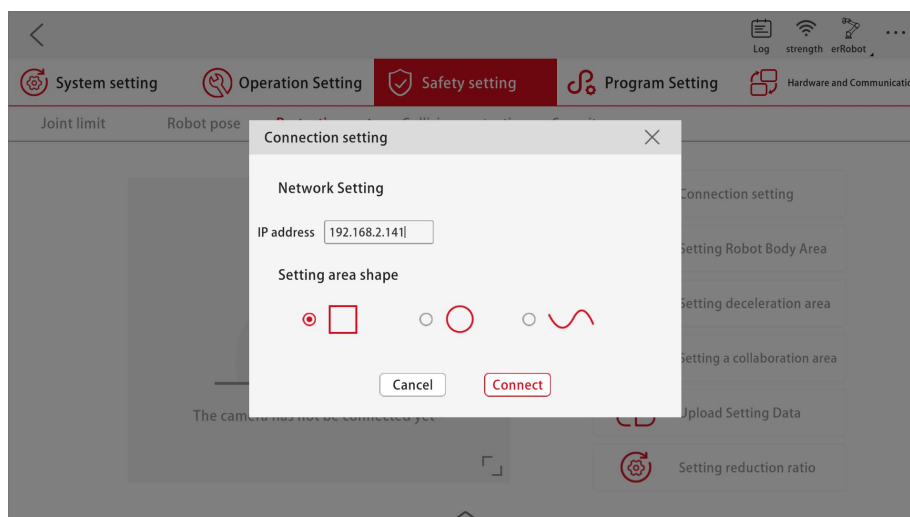


図 8 APP の接続設定

図 9 は、エリアの設定方法を示しています。当該モジュールに設定する必要があるエリアは以下三つが含まれます：ロボット本体エリア、減速エリア、協働エリア。三つエリアの働きはそれぞれ：

- ロボット本体エリア

名のとおり、ロボット本体が占めるエリアのことです。実際のプログラム実行時にはロボットが動いているので、ロボットを侵入物として検出しないためには、ロボットの活動エリアをアルゴリズムから除外する必要があります。当該モジュールでは、ロボット本体のエリアは円形を描く方法でしか決められません。設定方法：ロボットが実際に実行するジョブプログラムを実行し、視覚保護設定画面にてロボット本体エリアを設定し、ロボットが常時設定された本体エリアの円形内にいることを確認します。

● 減速エリア

当該エリアは、第一級保護エリアです。物体が当該エリアに侵入してくると、視覚保護装置がコントローラに情報を発信し、コントローラはロボットを縮小モードに制御します。当該エリアの設定はユーザーが設定時に選択した形状に基づいて実行します。下図のように、設定時には長方形の設定モードを選んだので、カスタマイズ減速エリアの形状も長方形になります。エリアを選んだ後、設定データアップロードをクリックして、設定情報を CPU に送るとともにローカルに保存します。アップロードしてエリア設定を一度に有効すると、次回に視覚保護システムの起動にもそのまま反映されます。エリア設定を変更したい場合は、再度設定し直してアップロードすればよい。

● 協働エリア

当該エリアは、第二級保護エリアです。物体が当該エリアに侵入してくると、視覚保護装置がコントローラに情報を発信し、コントローラはロボットを保護的停止モードに制御します。当該エリアの設定はユーザーが設定時に選択した形状に基づいて実行します。下図のように、設定時には長方形の設定モードを選んだので、カスタマイズ減速エリアの形状も長方形になります。エリアを選んだ後、設定データアップロードをクリックして、設定情報を CPU に送るとともにローカルに保存します。アップロードしてエリア設定を一度に有効すると、次回に視覚保護システムの起動にもそのまま反映されます。エリア設定を変更したい場合は、再度設定し直してアップロードすればよい。





図 9 APP カスタマイズエリア設定

以上は、視覚保護機能を実現するための重要な 2 つのステップを説明しましたが、これからは実際の操作における具体的な手順を説明します。:

- 視覚保護装置(ネットワークケーブルで POE 電源に接続)をオンします。
- PAD で対応する無線ネットワークに接続して APP を開き、ロボットに接続します。
- APP を使って保護エリアをカスタマイズ設定し、必要に応じてロボットの本体エリアを確認するために、まずジョブプログラムを実行してみます。
- ロボット電源を切ってから一旦切り、拡張 IO を設定します。
- 拡張 IO を実行状態に設定し、この時、システムがすでに視覚安全保護状態に入っています。
- ロボット電源を入れて起動し、ジョブプログラムを実行します。

5 注意事項

- **カスタマイズエリアの設定時には、干渉がないことを確認してください。**

このアルゴリズムでは、設定時のライブ画像を参照画像として使用し、視覚的な安全性を確保しています。従い、カスタマイズエリアを設定する際には、当該エリアに他の物体が干渉していないこと、即ち現時点に物体侵入がない状態にしておく必要があります。

- **カスタマイズエリア設定のロジックが正しいことを確保します。**

協働エリアは減速エリアよりも保護レベルが高いため、カスタマイズエリアの設定を行う際は、必ず協働エリアが減速エリアの範囲内にあることを確認してください。カスタマイズ曲線を使ってエリア設定を行う場合、描かれた曲線が極力閉鎖したエリアになるようにする必要があります。

- **視覚保護装置が正しく稼働している状態**

視覚保護装置はイーサネット POE によって電力供給されているため、イーサネットネットワークポートの状態を観察することで、視覚保護状態の動作状態を判断することができます。視覚保護装置のネットワークケーブルを POE 電源に差し込み、10 秒ほど経過して黄色のランプが常時点滅している状態となると、ネットワークポートが正常です。黄色いランプが断続的に点滅する場合は、ネットワークの状態が異常を表しており、視覚保護装置が正しく作動していません。POE 電源とネットワークケーブルが視覚保護装置に適しており、装置が正常に動作することを確認してください。

- **視覚保護機能のオンとオフ**

ネットワークケーブルが接続された後、コアプロセッサは視覚保護プログラムの実行を開始し、APP で拡張 IO がランモードに設定されている場合にのみ、視覚保護機能がロボットで有効になります。視覚保護機能をオフにしたい場合は、拡張 IO を編集に設定するか、ネットワークケーブルを切断することで、視覚保護プログラムの実行を停止してください。

- **視覚保護装置の取付高さ**

理論的には、視覚保護装置の取り付け高さが高いほど、設定可能な保護エリアは大きくなります。従って、設置高さは実際のニーズに応じて決定する必要があります。視覚保護装置が地上から 3m 離れている場合、保護エリアはおよそ 5m*4m になります。

付録六 Modbus Address シート

ID	タイプ	名称	データタイプ	機能コード	レジスタタイプ
8	汎用デジタル入力	DO0	BOOL	02	分離入力は読み取り可能であり、書き込み可能ではありません
9		DO1			
10		DO2			
⋮		⋮			
133		DO125			
134		DO126			
135		DO127			
40	汎用デジタル出力	DI0	BOOL	01/05/15	コイル状態
41		DI1			
42		DI2			
⋮		⋮			
165		DI125			
166		DI126			
167		DI127			
96	アナログ入力	AO00	UINT16	04	入力レジスタは読み取り可能であり、書き込み可能ではありません
97		AO01			
98		AO02			
99		AO03			
⋮		⋮			
109		AO13			
110		AO14			
111		AO15			
112		AO16	INT16		
113		AO17			
114		AO18			
⋮		⋮			
125		AO29			
126		AO30			
127		AO31			
128		AO32			
129					
130		AO33			
131					
132		AO34			
133					
⋮		⋮			
⋮					
186		AO61			
187					
188		AO62			
189					
190		AO63			

191					
-----	--	--	--	--	--

100	アナログ出力	AI00	UINT16	03/06	保持レジスタは読み取りと書き込みが可能です
101		AI01			
102		AI02			
103		AI03			
104		AI04			
⋮		⋮			
111		AI11			
112		AI12			
113		AI13			
114		AI14			
115		AI15			
116		AI16	INT16		
117		AI17			
118		AI18			
119		AI19			
120		AI20			
⋮		⋮			
127		AI27			
128		AI28			
129		AI29			
130		AI30			
131		AI31			
132		AI32	FLOAT32（ビッグエンディアンディスプレイ）		
133					
134		AI33			
135					
136					
137		AI34			
138					
139		AI35			
140					
141		AI36			
⋮	⋮				
⋮					
186	AI59				
187					
188	AI60				
189					
190	AI61				
191					
192	AI62				

193					
194					
195		AI63			

ID	タイプ	名称	データタイプ	機能コード	説明	単位	レジスタタイプ
300	ロボットデータ	サーボバージョン番号	INT32	04	/	/	入力レジスタは読み取り可能であり、書き込み可能ではありません
302		ロボットのシリアルナンバー					
304		関節 1 電圧 関節 2 電圧 関節 3 電圧 関節 4 電圧 関節 5 電圧 関節 6 電圧	INT32		各関節の電圧	V	
306							
308							
310							
312							
314							
316		関節 1 温度 関節 2 温度 関節 3 温度 関節 4 温度 関節 5 温度 関節 6 温度	INT32		各関節の温度	℃	
318							
320							
322							
324							
326							
328		関節 1 サーボエラーコード 関節 2 サーボエラーコード 関節 3 サーボエラーコード 関節 4 サーボエラーコード 関節 5 サーボエラーコード 関節 6 サーボエラーコード	INT32		各関節的サーボ序列号		
330							
332							
334							
336							
338							
340		関節 1 エラー状態 関節 2 エラー状態 関節 3 エラー状態 関節 4 エラー状態 関節 5 エラー状態 関節 6 エラー状態	UINT16		現在サーボエラー状態 0 はエラーがない 1 はエラーがある	/	
341							
342							
343							
344							
345							
346		関節 1 起動状態 関節 2 起動状態 関節 3 起動状態 関節 4 起動状態 関節 5 起動状態 関節 6 起動状態	UINT16		現在サーボ起動状態 0 は未起動 1 は起動		
347							
348							
349							
350							
351							
352	関節 1 衝突状態 関節 2 衝突状態 関節 3 衝突状態 関節 4 衝突状態	UINT16	現在サーボ衝突検出状態 0 は衝突がない 1 は衝突がある				
353							
354							
355							

356		関節 5 衝突状態					
357		関節 6 衝突状態					
358		関節 1 電流					
360		関節 2 電流					
362		関節 3 電流					
364		関節 4 電流					
366		関節 5 電流					
368		関節 6 電流					
370		センサートルク x					
372		センサートルク y					
374		センサートルク z					
376		センサートルク rx					
378		センサートルク ry					
380		センサートルク rz					
382		関節 1 位置					
384		関節 2 位置					
386		関節 3 位置					
388		関節 4 位置					
390		関節 5 位置					
392		関節 6 位置					

394		シャットダウン 1 スピード					
396		シャットダウン 2 スピード					
398		シャットダウン 3 スピード					
400		シャットダウン 4 スピード					
402		シャットダウン 5 スピード					
404		シャットダウン 6 スピード					
406		TCP 位置 X					
408		TCP 位置 Y					
410		TCP 位置 Z					
412		TCP 位置 RX					
414		TCP 位置 RY					
416		TCP 位置 RZ					
418		TCP スピード X					
420		TCP スピード Y					
422		TCP スピード Z					
424		TCP スピード RX					
426		TCP スピード RY					
428		TCP スピード RZ					
430		TCP_OFFSET_X					
432		TCP_OFFSET_Y					
434		TCP_OFFSET_Z					
436		TCP_OFFSET_RX					
438		TCP_OFFSET_RY					
440		TCP_OFFSET_RZ					

442	BASE_OFFSET_X		ユーザー座標系	mm	
444	BASE_OFFSET_Y				
446	BASE_OFFSET_Z				
448	BASE_OFFSET_RX				
450	BASE_OFFSET_RY				
452	BASE_OFFSET_RZ				
454	PROTECTIVE_STOP	UINT16	ロボット衝突が起きた：1 ロボット衝突が起きていない：0	°	
455	EMERGENCY_STOP		緊急停止		
456	POWER_ON		給電		
457	ROBOT_ENABLE		起動		
458	ON_SOFT_LIMIT		ソフトリミット		
459	INPOS		目標位置に到達		
460	運動モード		サーボ位置モード：4 アドミタンス制御モード：2 ドラッグモード：1 他のモード(Jog 等の操作)：0		
461	縮小モードレベル		縮小モードレベル： 一級縮小：1 二級縮小：2 保護性停止：3		
462	スピード倍率		FLOAT32		スピード倍率
464	MOTION_ERRCODE		INT32		エラーコード
466	CAB_TEMPERATURE	FLOAT32	制御盤温度		
468	CAB_AVERAGEPOWER		制御盤パワー		
470	CAB_AVERAGECURRENT		制御盤電流		
472	UHI_PULES	FLOAT32	コンベアパルス		
474	UHI_SPEED		コンベア運動スピード		
476	UHI_DIR	UINT16	コンベア運動方向		
477	UHI_ORIGIN_PULSES	INT32	コンベアオリジナルパルス		
479	保留	UINT16	/		