ABB

ROBOTICS

Application manual
Additional axes

Application manual

Additional axes

RobotWare 7.18

Document ID: 3HAC082287-001

Revision: G

Table of contents

Table of contents

Overview of this manual

About this manual

This manual details the setup of additional axes for the OmniCore controller.

Usage

This manual can be used as a brief description of how to install, configure and tune additional axes. It also provides information about related system parameters. Detailed information regarding system parameters, RAPID instructions and so on can be found in the respective reference manual.

Who should read this manual?

This manual is primarily intended for advanced users and integrators.

Prerequisites

The reader should...

- be familiar with industrial robots and their terminology
- be familiar with controller configuration and setup
- be familiar with the mechanical and dynamic properties of the controlled mechanism.

References

Reference	Document ID
Application manual - Controller software OmniCore	*3HAC066554-001*
Application manual - MultiMove	*3HAC089689-001*
Application manual - Functional safety and SafeMove	*3HAC066559-001*
Application manual - Servo Gun Setup	*3HAC086084-001*
Operating manual - RobotStudio	*3HAC032104-001*
Operating manual - OmniCore	*3HAC065036-001*
Operating manual - Integrator's guide OmniCore	*3HAC065037-001*
Technical reference manual - RAPID Instructions, Functions and Data types	*3HAC065038-001*
Technical reference manual - System parameters	*3HAC065041-001*
Product manual - OmniCore C90XT Type A	*3HAC089065-001*
Product manual - OmniCore V250XT Type B	*3HAC087112-001*
Product manual - OmniCore V400XT	*3HAC081697-001*
Application manual - TuneMaster	*3HAC063590-001*

Revisions

Revision	Description
A	Released with RobotWare 7.10.
B	Released with RobotWare 7.12. • Minor corrections.

Continues on next page

Revision	Description
C	Released with RobotWare 7.13. • Support for more than 1 ADU is introduced, for OmniCore V250XT Type B and OmniCore V400XT. • The brake current can now be seen in the device tree. • Added information about PTC supervision.
D	Released with RobotWare 7.14. • Added support for track motion.
E	Released with RobotWare 7.15. • Added support for MultiMove.
F	Released with RobotWare 7.16. • Added information for OmniCore C90XT Type A. • Information about power sharing and low power drive systems added in *Robot on page 86*, *Single on page 90* and *Configuration of the drive system on page 97*.
G	Released with RobotWare 7.18. • Added section *Coordinated axes on page 42*. • Minor corrections.

1 Introduction

1.1 Overview

Purpose

The additional axes option is used when the robot controller needs to control additional axes besides the robot axes. These axes are synchronized and, if desired, coordinated with the movement of the robot, which results in high speed and high accuracy.

When the controller is used in a robot system with external axes or a non-ABB manipulator, the system requires configuration and tuning as detailed in this manual. This manual can also be useful when such a system needs to be upgraded.

As external axes and non-ABB robots consume more power the drive system needs a more powerful transformer, rectifier and capacitor. In addition, suitable drive units must be installed in the controller. The hardware setup must also be configured with software to make the system functional.

Basic approach

This is the basic approach to set up additional axes in the robot controller.

- Installation
- Configuration
- Tuning

For a detailed description of how this is done, see the respective section.

For more information on the hardware components see *Hardware on page 97*.

> ⚠️ **WARNING**
>
> The manual mode peripheral speed of the external axis must be restricted to 250mm/s for personal safety reasons. The speed is supervised at three different levels, which means that three system parameters need to be set up. For more information see *Limit peripheral speed of external axis on page 19*.

1.2 Definitions

Robot

A robot is a mechanical unit with a tool center point (TCP). A robot can be programmed both in Cartesian coordinates (x, y and z) of the TCP and in tool orientation.

Single-robot system

A single-robot system can have

- only one motion task
- only one robot
- up to 6 additional axes (which can be grouped in an arbitrary number of mechanical units)
- up to 12 axes in total

> 💡 **Tip**
>
> In a single-robot system, semi-independent programming of individual mechanical units or axes can be achieved through the option *Independent Axes*. However, MultiMove is normally preferred when independent programming is desired.

MultiMove system

A MultiMove system can have

- up to 7 motion tasks
- up to 3 robots (3*6=18 axes)
- up to 24 axes, including the maximum 18 robot axes

Regardless of controller combination the limit is set to 3 IRB = 18 axis + total 6 ADU. These ADUs can be used placed in any combination of controllers. If all 6 are used in one V400XT then no ADU can be used in other controllers.

See *Application manual - MultiMove*.

Additional axes

The robot controller can control additional axes besides the robot axes. They can be jogged and coordinated with the movements of the robot. The system may have a single additional axis, for example, a motor, or a set of additional axes such as a two axis positioner.

1.3 General guidelines and limitations

Use integer gear ratio

The transmission gear ratio between motor and arm of a continuously rotating axis shall be an integer in order not to cause calibration problems when updating revolution counters.

When the revolution counter is updated, the number of motor revolutions is reset. In order for the zero position of the motor to coincide with the zero position of the arm, independent of number of revolutions on the arm side, the gear ratio needs to be an integer (not a decimal number).

Example: Gear ratio = 1:81 (not 1:81.73).

This problem will only be visible when updating revolution counters with the arm side rotated n turns from the original zero position. I.e. an axis with mechanical stops will not have this problem.

This page is intentionally left blank

2 Getting started

2.1 Get started with additional axes, servo guns

Overview

This section describes the steps to get started with:

- additional axes
- a servo gun

Step by step

	Action	See
1	Install the hardware, such as motor unit, connection box cables and connectors.	See the product manual for the respective product.
2	If a brake is used, connect the brake to the drive unit for the additional axis.	*Brakes on page 102*.
3	Install the RobotWare software and modify the system using RobotStudio. Depending on what additional axis is used, there are specific Add-Ins in RobotStudio for parts of the configuration.	How to modify the system for motor units, gear units, positioners, and track motion from ABB is described in the respective product manual.
4	For MultiMove, make sure to connect the additional axis to the correct controller.	MultiMove configuration is described in *Application manual - MultiMove*.
5	Download the system to the robot controller.	
6	For additional axes and servo guns there are various template files available depending on the setup of the hardware. If the user does not already have a specific template file, see information on what file to use.	*Template files on page 16*.
7	Use RobotStudio to configure the system parameters.	*Limit peripheral speed of external axis on page 19*. *Minimal configuration of general additional axes on page 21*. *Minimal configuration of servo gun on page 23*.
8	Verify if any advanced setting needs to be done.	*Advanced settings on page 26*.
9	When the configuration is done, continue with fine calibration and tuning.	*Commutation with service routine on page 54*. *Tuning of servo control parameters on page 59*.

This page is intentionally left blank

3 Installation

3.1 Standard additional axis

Overview

Normally all necessary configuration parameters regarding drive unit and power units are pre-loaded by ABB, and do not need to be re-installed. For more information on how to add options to the system, see *Operating manual - RobotStudio* or *Operating manual - Integrator's guide OmniCore*.

Peripheral equipment

If the supplier of other equipment, such as track motion or peripheral equipment, supplies configuration files, these files should be used instead of the standard files. For more information, see the documentation provided by the supplier.

15

3.2 Template files

Overview

This section details the template files for respective hardware. Normally you only need to change the motor data in these files. For more information on how to change these files, see *Operating manual - RobotStudio*.

The template files are located in the following directory in the RobotWare distribution:

...\RobotPackages\RobotControl_<version>\utility\AdditionalAxis\General.

> 🔆 **Tip**
>
> Navigate to the RobotWare installation folder from the RobotStudio **Add-Ins** tab, by right-clicking on the installed RobotWare version in the **Add-Ins** browser and selecting **Open Package Folder**.

Motors for additional axis

There are template files that can be used to connect the motors to the drive system.

The following table shows available template files for motors connected to drive module 1. They are located in:

...\RobotPackages\RobotControl_<version>\utility\AdditionalAxis\General.

Similar template files exist for drive modules 2-6. These files are adjusted for additional axes on the same drive unit as a robot.

File name	Measurement node
M7_N1_DM1.cfg	7
M7_DM1.cfg	1

Servo gun

The template files for servo gun are prepared for drive unit 1-6. The files contain default data for servo gun. Motor data etc. for selected motor must be changed. Listed below are the template files for cabinet 1.

Servo gun template files located in:

...\RobotPackages\RobotControl_<version>\utility\AdditionalAxis\ServoGun.

File name	Measurement node
M7N7S_DM1.cfg	1
M7_DM1.cfg	7
M8_DM1.cfg	2

Continues on next page

Track motion

The template files for track motion are prepared for drive unit 1-6. The files contain default data for track motion. Motor data etc. for selected motor must be changed. Listed below are the template files for cabinet 1.

Track motion template files located in: *...\utility\AdditionalAxis\Track*.

File name	Measurement node
M7N7S_DM1.cfg	1
M7_DM1.cfg	7
M8_DM1.cfg	2

Recommended combinations

The following combination of configuration files for motor 7, 8, and 9 are the recommended combinations in one drive module.

Motor 7	Motor 8	Motor 9
M7_DM1.cfg	M8_DM1.cfg	M9_DM1.cfg
M7_DM1.cfg	M8_DM1.cfg	M9_DM1.cfg
M7_DM1.cfg	M8_DM1.cfg	M9_DM1.cfg

This page is intentionally left blank

4 Configuration

4.1 Basic settings

4.1.1 Limit peripheral speed of external axis

> ⚠️ **CAUTION**
>
> Incorrectly defined parameters will result in incorrect speed. Always verify the speed after changing these parameters.
>
> There is a hazard that the speed 250 mm/s is exceeded in manual reduced speed mode.

Calculate parameter values

Two system parameters need to be configured. The parameters belong to the type *Supervision Type* in the configuration topic *Motion* and are expressed in ratio of max speed (1 = 100%).

Teach Max Speed Main

Teach Max Speed Main = (x / *Arm Length*) * (*Transmission Gear Ratio* / *Speed Absolute Max*)

where:

- x is the speed in mm/s
- *Transmission Gear Ratio* belongs to the type *Transmission*.
- *Speed Absolute Max* belongs to the type *Stress Duty Cycle* (rad/s).
- *Arm Length* should be measured from the rotational center of the external axis (meter).

If the result of the calculation exceeds 0.94, use 0.94 instead of the calculated value.

Insert the calculated result at the type *Supervision Type: Teach Max Speed Main*.

Teach Max Speed DSP

Calculate and use the largest value of:

- *Teach Max Speed Main* * 1.20
- *Teach Max Speed Main* + (8 / *Speed Absolute Max*)

Insert the calculated result at the type *Supervision Type*: *Teach Max Speed DSP*.

Example

Given parameter values

Transmission Gear Ratio = 120

Speed Absolute Max = 320 rad/s

Arm Length = 0.5 m

Calculations

Teach Max Speed Main = (0.25 /*Arm Length*) * (*Transmission Gear Ratio* / *Speed Absolute Max*) = (0.25 / 0.5) * (120 / 320) = 0.188

Teach Max Speed Dsp = max{(*Teach Max Speed Main* * 1.20) , (*Teach Max Speed Main* + (8 / *Speed Absolute Max*))} = max{(0.188 * 1.2) , (0.188 + (8 / 320))} = max{0.226 , 0.213} = 0.226

4.1.2 Minimal configuration of general additional axes

Overview

This section describes how to make a minimal configuration of a standard additional axes.

> ⚠️ **WARNING**
>
> Incorrect definition of system parameters for additional axes may cause damage to the robot or personal injury.

Load parameters

Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*.

	Action
1	Right click on configuration icon in the system view, and select **Load Parameters**.
2	Select **Load parameters if no duplicates** and click **Open**.

3	Browse to the template files in the RobotWare installation, see *Template files on page 16*.
	• For general additional axis, browse to the directory:
	   ...\RobotPackages\RobotControl_<version>\utility\AdditionalAxis\General
	• For track motion, browse to the directory:
	   ...\RobotPackages\RobotControl_<version>\utility\AdditionalAxis\General\Track

4	Select the configuration file for required axes and click **Open**.
5	Perform a warm start of the system from the FlexPendant or RobotStudio.

Configure parameters

Use RobotStudio or the FlexPendant to perform the following instructions. See *Operating manual - RobotStudio*.

For parameter description, see *System parameters on page 65*.

	Action	Info/Illustration
1	Select the topic *Motion* and type *Mechanical Unit*, and define the following in the parameter. ℹ️ **Note** For a single axis mechanical unit without kinematic model, *Name* and *Use Single 1* in the type *Mechanical Unit* and *Name* in the type *Single* must be the same.	• *Name* • *Activate at Start Up* • *Deactivation Forbidden* • *Use Single 1* • *Allow Move of User Frame*
2	Select the topic *Motion* and type *Single* and specify which *Single Type* to use.	• *Name* • *Use Single Type*
3	Select the topic *Motion* and type *Single Type* and specify the type of additional axis in the parameter *Mechanics*.	Example of values of the parameter *Mechanics*: • TRACK • FREE_ROT

4 Configuration

	Action	Info/Illustration
4	Select the topic *Motion* and type *Joint* and set the parameter *Logical Axis* to the logical axis number.	Example: Logical axis 10 will then correspond to the field `eax_d` in the RAPID data type `robtarget`.
5	Select the topic *Motion* and type *Arm* and specify the arm characteristics for the axis.	• *Upper Joint Bound* • *Lower Joint Bound*
6	Select the topic *Motion* and type *Acceleration Data* and specify the arm performance for the axis.	• *Nominal Acceleration* • *Nominal Deceleration*
7	Select the topic *Motion* and type *Transmission* and specify the following.	• *Transmission Gear Ratio* • *Rotating Move* • *Transmission High Gear* • *Transmission Low Gear*
8	Select the topic *Motion* and type *Motor Type* and specify the following.	• *Pole Pairs* • *ke Phase to Phase* (Vs/rad) • *Max current* (A rms) • *Phase resistance* (ohm) • *Phase inductance* (H)
9	Select the topic *Motion* and type *Motor Calibration* and define the calibration and commutation offsets.	• *Calibration Offset* • *Commutator Offset*
10	Select the topic *Motion* and type *Stress Duty Cycle* and define the torque and speed absolute max on the motor side.	• *Torque Absolute Max* (Nm) • *Speed Absolute Max* (rad/s)
11	Perform a warm start of the system.	

> **Note**
>
> If *Torque Absolute Max* is too high it may result in a configuration error at restart.

Limitations

If *Torque Absolute Max* is too high it may result in a configuration error at restart. To avoid errors, do not set *Torque Absolute Max* higher than:

Torque Absolute Max $< \sqrt{3}$ * *ke Phase to Phase (Vs/rad)* * *Max Current*

where:

- *Max Current*, belonging to the type *Motor Type*, is the maximum current of the used drive module
- *ke Phase to Phase (Vs/rad)*, belonging to the type *Motor Type*, is a voltage constant

4.1.3 Minimal configuration of servo gun

Overview

> ⚠️ **WARNING**
>
> Incorrect definition of system parameters for brakes or additional axes may cause damage to the robot or personal injury.

Load parameters

Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*.

	Action
1	Right click on configuration icon in the system view, and select **Load Parameters**.
2	Select **Load parameters if no duplicates** and click **Open**.
3	Browse to the template files in the RobotWare installation, see *Template files on page 16*. *...\RobotPackages\RobotControl_<version>\utility\AdditionalAxis\ServoGun*
4	Select the configuration file for required axes and click **Open**.
5	Perform a restart of the system from the FlexPendant or RobotStudio.

Configure parameters

Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*.

For parameter description, see *System parameters on page 65*.

	Action	Info/Illustration
1	Select the topic *Motion* and the type *Mechanical Unit* and define the following parameter:	• *Name*
2	Select the topic *Motion* and the type *Joint* and specify the logical axis number under parameter *Logical Axis*.	Example: Logical axis 10 will then correspond to the field `eax_d` in a RAPID data of the type `robtarget`.
3	Select the topic *Motion* and the type *Arm* and specify the arm characteristics for the axis.	• *Upper Joint Bound* • *Lower Joint Bound* *Lower Joint Bound* should be set to zero or a small negative value (e.g. -0.005 m) in order to protect the gun from collisions. The limit is not active during force control of the gun. For force control there is another positional limit, *Max Force Control Position Error*, in the type *Supervision*.
4	Select the topic *Motion* and the type *Acceleration Data* and specify the arm performance for the axis.	• *Nominal Acceleration* • *Nominal Deceleration*
5	Select the topic *Motion* and the type *Transmission* and specify the following parameters:	• *Transmission Gear Ratio*

4 Configuration

	Action	Info/Illustration
6	Select the topic *Motion* and the type *Motor Type* and specify the following parameters:	• *Pole Pairs* • *ke Phase to Phase* (Vs/rad) • *Max Current* (A rms) • *Phase Resistance* (ohm) • *Phase Inductance* (H)
7	Select the topic *Motion* and the type *Motor Calibration* and define the calibration and commutation offsets.	• *Calibration Offset* • *Commutation Offset*
8	Select the topic *Motion* and type *Stress Duty Cycle* and define the torque and speed absolute max on the motor side.	• *Torque Absolute Max* (Nm) • *Speed Absolute Max* (rad/s)
9	Perform a restart of the system from the FlexPendant or RobotStudio.	

Tuning

After configuration additional axis tuning needs to be performed. See *Tuning of servo control parameters on page 59* for tuning of the system.

Configure servo gun parameters

After tuning, the servo gun specific parameters can be defined.

	Action	Info/Illustration
1	Select the topic *Motion* and the type *Supervision Type* and define the supervision limits during force control.	• *Max Force Control Position Error* • *Max Force Control Speed Limit*
2	Select the topic *Motion* and the type *SG Process* and define the process parameters specific for servo gun.	• *Sync check off* • *Close Time Adjust* • *Force Ready Delay* • *Max Force Control Motor Torque* • *Post-synchronization Time* • *Calibration Mode* • *Calibration Force High* • *Calibration Force Low* • *Calibration Time*

Continues on next page

Tip force

The relationship between the programmed tip force and the resulting motor torque is set up in the following parameters. The torques may be negative due to the sign of the gear ratio while the forces must always be positive. Before setting up this table, the parameters in the Force Master should be tuned. See *Application manual - Servo Gun Setup*.

The easiest way to set up the table is by using a RAPID force calibration service routine.

Parameter	Description
Number of Stored Forces	Number of stored forces in the force vs motor torque table. The minimum value allowed is 2.
Tip Force 1	Gun tip force 1 (N)
Motor Torque 1	Motor torque 1 (Nm)
Tip Force 2	Gun tip force 2 (N)
Motor Torque 2	Motor torque 2 (Nm)
Tip Force 10	Gun tip force 10 (N)
Motor Torque 10	Motor torque 10 (Nm)

4.2 Advanced settings

4.2.1 Disconnect a servo motor

Overview

It is possible to disconnect and reconnect the motor of a deactivated axis if a certain deactivation mode is set up.

> ℹ️ **Note**
>
> If the axis is moved when disconnected, the position of the axis might be wrong after reconnecting, and this will not be detected by the controller. The position after reconnection will be correct if the axis is not moved, or if the movement is less than 0.5 motor revolutions. For servo guns, there is a RAPID calibration method available (the ToolChange calibration) that will adjust any positional error caused by gun movement during disconnection.

Configure parameters

Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*

For parameter description, see chapter *System parameters on page 65*.

	Action	Info/Illustration
1	Select the topic *Motion* and type *Measurement Channel* and define the following in the parameter.	• *Disconnect at Deactivate*

4.2.2 Servo Tool Change

Overview

With the option *Servo Tool Change* it is possible to disconnect the resolver and motor cables from the motor of one external axis and connect them to the motor of another additional axis.

For details about *Servo Tool Change*, see *Application manual - Controller software OmniCore*.

> ⚠️ **WARNING**
>
> It is important that no other mechanical unit used with one tool changer are activated, but the one corresponding to the currently connected servo gun! An activation of the wrong mechanical unit may cause unexpected movements and personal injury.

Considerations

The list below specifies special considerations when switching motors:

- The two (or more) additional axes sharing the same motor cables are configured as separate mechanical units.
- The additional axes are configured to use the same measurement node and drive unit node.
- If two servo guns are used with a tool changer, the template file `M7L1B1S_DM1.cfg` can be used for configuration of both guns (change the name of the instance in one of the files).
- A motor switch can be done only if all sharing axes are deactivated.
- The reconnected motor is activated and this activation will restore the position of the axis to the latest position.
- Always use the tool change tip calibration after activation (for spot welding guns).

Connection relay

To make sure that the correct mechanical unit is active, some tool changers support I/O signals that specify which gun is currently connected.

It is also possible to lock unconnected mechanical units from activation by specifying a connection relay and connect it to a digital input (DI).

Defining a connection relay

Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*.

For parameter descriptions, see *System parameters on page 65*.

	Action	Parameter
1	Select the topic *Motion* and type *Mechanical Unit* and define the name of the relay, or check the name if this is already defined.	• *Use Connection Relay*

Continues on next page

4.2.2 Servo Tool Change
Continued

	Action	Parameter
2	Select the topic *Motion* and type *Relay* and select the named relay, if this is defined.	
3	If the named connection relay is not defined, a new relay must be created.	
4	Change the name of the newly created relay to the same as the *Use Connection Relay* parameter. Define an activation lock signal.	• *Name* • *Output Signal* • *Input Signal*

4.2.3 Defining relays

Overview

   The additional drive unit can be activated via signals from the robot controller. When a module is activated, e.g. by choosing the module in the Jogging window on the FlexPendant, the output signal is automatically set. A check is made later that the corresponding input signal from the relay is set.

Defining activating relays

   Define the input and output signals for all connected relays. Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*. For parameter description, see *System parameters on page 65*.

	Action	Parameters
1	Restart the controller to check that the additional axes can be activated from the I/O window on the FlexPendant.	
2	Select the topic *Motion* and type *Relay* and define the following parameters.	• *Name* • *Output Signal* • *Input Signal*
3	Select the topic *Motion* and type *Mechanical Unit* and specify the name of the activation relay.	• *Use Activation Relay*
4	Perform a restart of the system.	

4.2.4 Defining brake behavior

Overview

If the axis has a brake, parameters which control brake behavior should be configured. If the axis is affected by gravity, more accurate parameter settings are necessary.

This section describes how to set up brake behavior for additional axes and non ABB robots.

Brake behavior at emergency stop

When an emergency stop has been ordered, it will take about 50 to 300 ms before the mechanical brake is physically active. Meanwhile there is ramp deceleration by motor. After a certain period of time, the speed of the axis will determine whether or not the electrical torque brake is to be used along with the mechanical brake.

Good brake behavior is characterized by low oscillation in speed during deceleration. TuneMaster can be used to verify this; study *Speed* (signal number 6) and *TorqueRef* (signal number 9).

Measures must be taken to prevent the axis from dropping due to gravitation. This will happen if the motor torque is turned off before the mechanical brake has become physically active.

Emergency brake algorithm

There are a few parameters which need to be configured to achieve good brake behavior. Understanding their role in the brake algorithm of the robot controller will simplify the task. The scenarios below illustrate how the brake parameters support the emergency brake algorithm of the robot controller.

Scenario 1

Axis almost at standstill after brake ramp, electrical torque brake not activated.
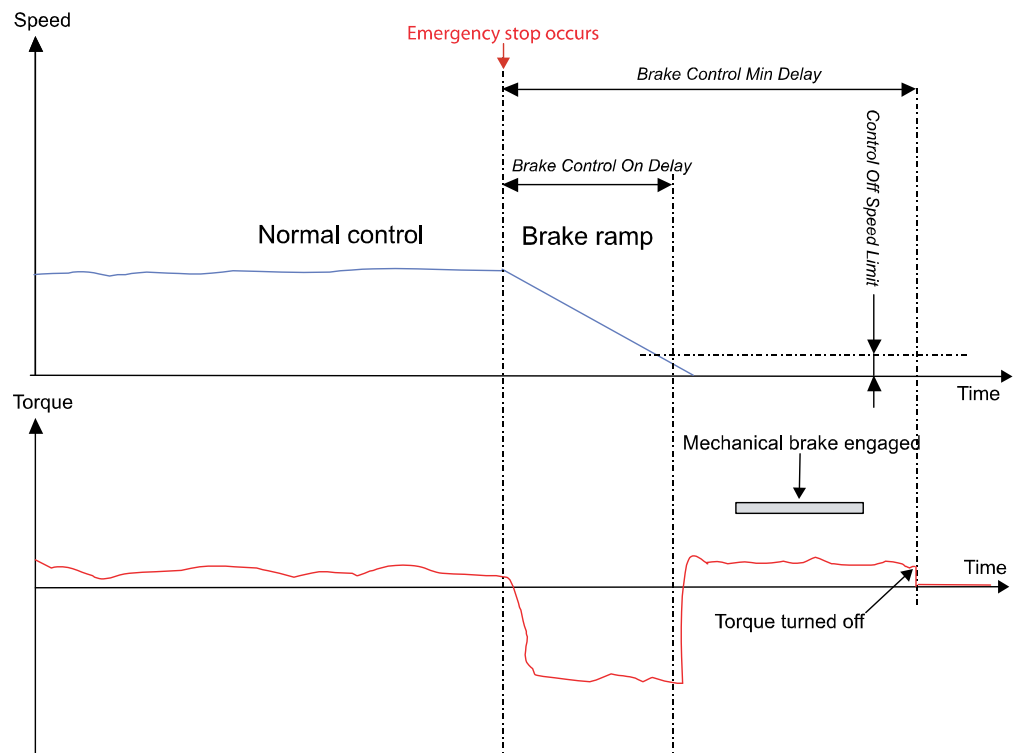
en0600003145

1 Emergency stops occurs. Immediately, ramp deceleration by motor is started.

2 The axis has stopped when the *Brake Control On Delay* time has passed. The motor torque is used until the time *Brake Control Min Delay* has passed. This prevents the axis from falling before the mechanical brake is engaged.

Scenario 2

Axis still moving after brake ramp, electrical torque brake activated.
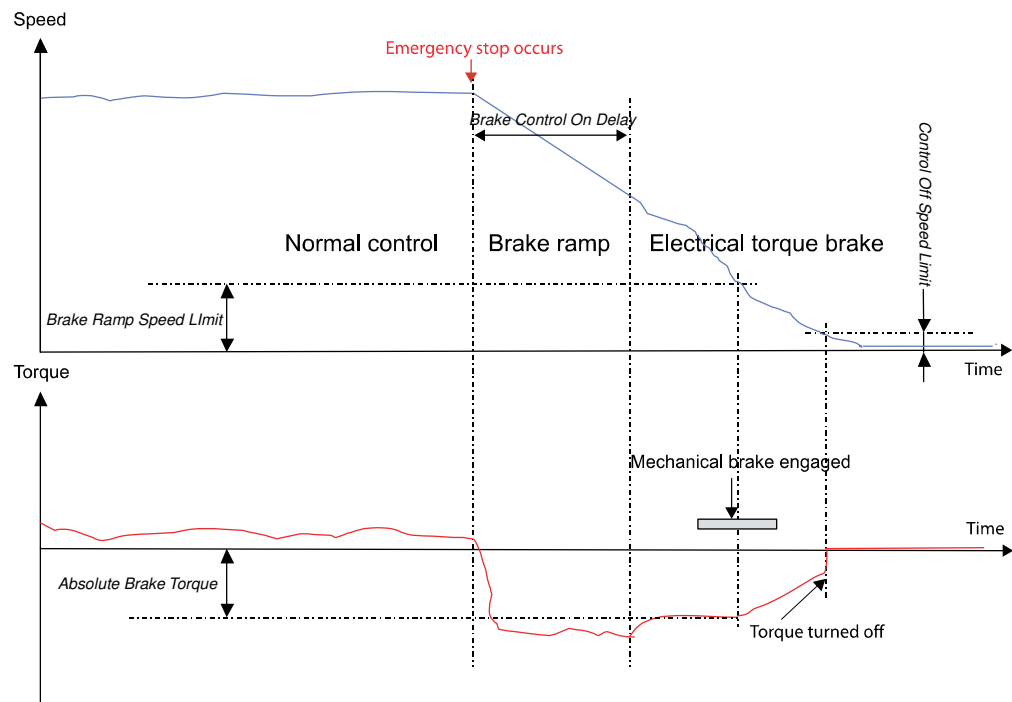
en0600003146

1. Emergency stops occurs. Immediately, ramp deceleration by motor is started.
2. As the axis is still moving (that is, speed exceeds *Control Off Speed Limit*) when *Brake Control On Delay* time has expired, the brake algorithm changes to Electrical torque brake.
3. The motor generates a brake torque specified by *Absolute Brake Torque*.
4. Torque reduction is started when the axis speed equals the value of *Brake Ramp Speed Limit*.
5. When the axis comes to a standstill the motor torque is turned off.

Defining brake parameters

Use RobotStudio or the FlexPendant to configure the brake parameters of the axis. See for recommended start values for some of these parameters. All parameters belong to the type *Brake* in the configuration topic *Motion*.

	Action	Note
1	Define *Control Off Delay* in seconds.	Specifies for how long the control of the axis should be active. Time should be longer than it takes for the mechanical brake to become physically active, as to prevent the axis from dropping due to gravitation.
2	Define *Brake Control Min Delay* in seconds.	Should be set to the same value as *Control Off Delay*.

Application manual - Additional axes
3HAC082287-001 Revision: G

	Action	Note
3	Define *Brake Control On Delay* in seconds.	Specifies the period of time during which deceleration by motor is used. It should be set close or equal to the mechanical brake activation time, but must be long enough to damp mechanical oscillation. After the time has expired, the speed of the axis is measured against *Control Off Speed Limit*. If it is higher the electrical torque brake is activated.
4	Define *Absolute Brake Torque* in Nm.	Specifies max brake torque generated by the motor in the electrical torque brake phase. *Absolute Brake Torque* together with torque generated by the mechanical brake must not exceed max allowed torque for the arm, in order not to damage arm and gearbox.
5	Define *Brake Ramp Speed Limit* in rad/s.	Specifies the speed limit for torque reduction in the electrical torque brake phase and is typically set to zero.
6	Perform a restart of the controller.	

Getting started with a simple brake configuration

To facilitate brake configuration, this section provides initial values for some brake parameters. It is necessary, however, to adjust these parameter settings until good brake behavior is achieved.

The table below shows recommended initial values. All parameters belong to the type *Brake* in the configuration topic *Motion*.

Parameter	Start value
Control Off Delay	150% of mechanical brake activation time
Brake Control Min Delay	same as Control Off Delay
Brake Control On Delay	mechanical brake activation time
Absolute Brake Torque	0
Brake Ramp Speed LImit	0

Note

Do not modify *Control Off Speed Limit*! Its predefined ratio of max speed value defines zero speed.

4.2.5 Supervision

Overview

Supervision is used to avoid overload on the motors. To prevent misleading supervision errors due to influence forces, all axes with mutual influences shall be configured to the same influence group.

Description

If a manipulator mounted on a track motion accelerates, the reaction (influence) forces affect the track motion. In the same way, if the track motion accelerates, the manipulator is affected. Up to 10 different influence groups can be used (1-10). By default the manipulator belongs to the influence group number 1.

> ⚠️ **WARNING**
>
> If PTC sensor supervision is required, this is activated by setting the system parameter *Enable additional axes ptc supervision* to true (topic *Motion*, type *Drive Module User Data*).

Define influence groups

Define the parameter for supervision on the additional axes. Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*. For parameter description see *System parameters on page 65*.

	Action	Parameters
1	Select the topic *Motion* and type *Supervision Type*.	
2	Select the additional axes to be grouped.	
3	Specify the following parameter. Default value: **0**.	*Influence Group*

4.2.6 Independent joint

Overview

With the option *Independent Axis*, an additional axis (linear or rotating) can run independently of the other axes of the robot.

Description

An axis is set in independent mode by executing an independent move instruction. Use the independent reset instruction to return to normal mode. Independent reset instruction can also be used in normal mode in order to change the logical position of the axis.

For more information about Independent Axis, see *Application manual - Controller software OmniCore*.

Activate Independent Joint

Define the parameter for supervision on the additional axes. Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*. For parameter description see section *System parameters on page 65*.

	Action	Parameter
1	Select the topic *Motion* and type *Arm*.	
2	Double click the axis to be activated	
3	Select the parameter *Independent Joint* in the appearing list.	
4	Set the Independent Joint to value **On**.	• *Independent Joint* • *Independent Upper Joint Bound* • *Independent Lower Joint Bound*

Defining transmission ratio

For external axes, the transmission ratio must be defined as normal with the parameter *Transmission Gear Ratio*, but also with its nominator and denominator values in order to get exact value (no rounding off). Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*. For parameter description see section *System parameters on page 65*.

	Action	Parameter
1	Select the topic *Motion* and type *Transmission*.	
2	Specify the following parameters.	• *Transmission High Gear* • *Transmission Low Gear*

4.2.7 Soft servo

Overview

Soft servo can be activated for additional axes which are configured with *Lag Control Master 0*. The behavior of movements with the soft servo activated is described in the *Technical reference manual - RAPID Instructions, Functions and Data types*.

Description

There are four system parameters to consider when the soft servo is used for an additional axis. The parameter are set to default values.

Set soft servo parameters

Define the parameter for soft servo on the additional axes. Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*. For parameter description see section, *System parameters on page 65*.

	Action	Parameters
1	Select the topic *Motion* and type *Lag Control Master 0*.	
2	Select the lag control master corresponding to the external axis.	
3	Select the desired parameter and change its value.	• *K Soft Max Factor* • *K Soft Min Factor* • *Kp/Kv Ratio Factor* • *Ramp Time*
4	Click **OK** to confirm.	
5	Perform a restart for the changes to take effect.	

4 Configuration

4.2.8 Activate force gain control

Overview

Force gain control is used in cases when heavy load, high friction and low speed makes it difficult for an additional axis to reach its end point.

All axes that affect force gain control must be within a certain position range from the end point before forced gain control is enabled. This position range is also specified in the configuration topic *Motion*, type *Supervision*.

Description

When activating forced gain control for an additional axis, two types under *Motion* must be considered. Decide which axes should have forced gain control in *Lag Control Master 0*, and decide which axes should affect forced gain control in *Supervision*.

Set force gain control parameters

Define the parameter for forced gain control on the additional axes. Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*. For parameter description see section, *System parameters on page 65*.

	Action	Parameter
1	Select the topic *Motion* and type *Lag Control Master 0*.	
2	Select the lag control master corresponding to the additional axis.	
3	Select the parameter to be changed.	• *Forced Control Active* • *Forced Factor for Kp* • *Forced Factor for Ki* • *Rise Time for Kp*
4	Press: **OK** to confirm.	
5	Perform a restart for the changes to take effect.	

Set supervision parameters

Define the parameter for supervision on the additional axes. Use RobotStudio to perform the following instructions. See *Operating manual - RobotStudio*. For parameter description see section, *System parameters on page 65*.

> ⚠ **CAUTION**
>
> Do not change supervision for the robot axes. Changes on these values could affect the service interval cycles and impair its performance.

	Action	Parameter
1	Select the topic *Motion* and type *Supervision*.	
2	Select the supervision corresponding to the axis that should or should not affect forced gain control.	

4.2.8 Activate force gain control
Continued

	Action	Parameter
3	Select the parameter to be changed.	• *Affects Forced Control* • *Forced on Position Limit* • *Forced off Position Limit*
4	Click **OK** to confirm.	
5	Perform a restart for the changes to take effect.	

4.2.9 Defining parameters for general kinematics

Overview

It is possible to use general kinematics for most manipulators. A set of template configuration files can be found in *...\utility\AdditionalAxis\DM1\GeneralKinematics.*, see *Application manual - Additional axes*.

> **i** **Note**
>
> Definition is not possible via the FlexPendant or RobotStudio. PC editing of the configuration files is necessary.

General kinematics for robots

The following needs to be defined.

Type	Description
ROBOT_TYPE	• base_pose_rot_u0, base_pose_rot_u1, base_pose_rot_u2, base_pose_rot_u3 (Rotation between user defined robot base and internal base according to Denavit - Hartenberg definition). • no_of_joints = highest joint number • type **GEN_KIN**
TRANSMISSION	For each arm of the additional robot in question. • **rotating_move** if rotating axes, exclude otherwise
ARM_TYPE	For each arm of the external robot in question. • **length** • **theta_home_position** • **offset_z** • **attitude** For information about the parameters, see *Application manual - Additional axes*.

4.2.10 Enabling *Service Information System* **functions**

Overview

To enable *Service Information System* functions such as *GetAxisDistance*, the following needs to be added manually to the configuration file.

This modification cannot be done in RobotStudio or on the FlexPendant.

Example for non-ABB robot

```
ROBOT_TYPE:
-name "STN1" -type "IRBP_A" -error_model "NOMINAL"\
-no_of_joints 2 -use_sis "STN1"
SIS:
-name "STN1" -sis_active -use_sis_joint_0 "STN1_sis_joint_0"\
-use_sis_joint_1 "STN1_sis_joint_1"
#
SIS_JOINT_INFO:
-name "STN1_sis_joint_0"
-name "STN1_sis_joint_1"
```

Example for single

```
SINGLE_TYPE:
-name "M7DM1" -mechanics "FREE_ROT" -use_sis "M7DM1"
SIS_SINGLE:
-name "M7DM1" -sis_active -use_sis_param "M7DM1"
```

4.2.11 Safe Disable of Drive Unit

Safe Disable of Drive Unit

Safe Disable of Drive Unit is a function that can be used to disable a manipulator or additional axis, by safely setting its drive unit in a state with all brakes applied and servo control switched off. This can, for example, be used to prevent unexpected start-up.

Functionality

The function is activated by a safe input signal. It shall be used when there is no movement on the mechanical unit. If the function is activated while one or more axes of the mechanical unit are moving, a violation is triggered with a stop category 0 or stop category 1, depending on the configuration.

Related information

More information about the safety function *Safe Disable of Drive Unit* is available in *Application manual - Functional safety and SafeMove*.

4.3 Coordinated axes

4.3.1 About coordinated axes

Additional axes, general

All additional axes are handled in mechanical units. This means that before an additional axis may be moved, the mechanical unit to which it belongs, must be activated. Within a mechanical unit, the different axes will be given a logical name, from *a* to *f*. In the system parameters, these logical axes will be connected to the additional axes joints. For each joint a motor and a drive unit is defined. Different joints may share the same motor and drive unit.

Two or more mechanical units may be activated at the same time, as long as they do not have the same logical axes defined in their set of additional axes. However, two or more mechanical units may have the same logical axes, if they are not activated simultaneously. Two or more mechanical units may not be activated at the same time, if they share one or more drive units, even if they use separate logical axes. This means that two logical axes, each belonging to different mechanical units, may control the same drive unit, but not at the same time.

Coordination

A mechanical unit may be coordinated or not coordinated with the robot movements. If it is not coordinated, each axis will be moved independent of the robot movements, e.g. when jogging, only the separate axis will move. However during program execution, the additional axes will be synchronized to the robot movement, in such a way that both movements will be completed in the same time.

If the mechanical unit is coordinated, the TCP velocity in the object coordinate system, will be the programmed velocity irrespective of the movements of the additional axes. Two types of coordination categories exist. The first category of coordination is when the robot base is moved, e.g. the coordination to a gantry or track movement. This means that the robot is mounted on a gantry or a track, and may be moved along these axes. The world and user/object coordinate systems, however, will be fixed in the room, and the robot movements in these coordinate systems will be independent of simultaneous gantry or track movements. This coordination is automatically active, if the mechanical unit with the track motion is active.

The second coordination category, is when the robot movements are coordinated to the movements of a user frame connected to a mechanical unit. E.g. a user frame may be placed on a turntable and connected to its movements. An ordinary work object may be used for this purpose, if it is marked with the name of the mechanical unit to be connected to, and that it should be moveable. The coordination will be active if the mechanical unit is active, and the coordinated work object is active. When such a coordinated work object is used, in jogging or in a move instruction, the data in the `uframe` component will be ignored and the location of the user coordinate system will only depend on the movements of the mechanical unit. However the `oframe` component will still work giving an object frame related to the user frame and also the displacement frame may be used.

4.3.2 Coordinated track motion

4.3.2.1 How to get started with a coordinated track motion

Coordination procedure

In the checklist below, the steps required to coordinate track motion are described. In each step, there may be a reference to another chapter in this manual, where more details of the specific actions to be taken will be found.

> **ℹ Note**
>
> When a robot is coordinated with 3rd party track, the performance of the track will be reduced to match limitations from the robot.

	Action	Info/illustration
1	Make sure the system parameter *Mechanics* in the type *Single Type* is set to TRACK.	
2	Calibrate the robot and the track motion, i.e. the zero position of the measuring system for both robot and track must be carefully determined.	See section *Calibrating* in *Operating manual - OmniCore*.
3	Define the base frame of the robot. This defines the robot base frame relative to the world frame. The procedure is necessary only if the world frame is separate from the robot base frame. Please observe that the track must be in its calibration position when the robot base frame is defined.	See section *4 points XZ calibration* in *Operating manual - OmniCore*.
4	Define the base frame of the track. This defines the rotation of the robot base relative to the track.	See *Defining the base frame for a track motion on page 45*.
5	Activate the base frame coordination by setting the system parameter *Base frame moved by* (topic *Motion* and type *Robot*) for the robot to the name of the track.	**⚠ WARNING** The combined speed of the track motion and the robot must not exceed 250 mm/s in manual mode. This can only be ensured if the correct mechanical unit (the track motion) is set in the parameter *Base frame moved by* for the robot.
6	Create a backup of the system by tapping **Settings - Backup & Restore - Backup**.	See *Back up and restore systems* in *Operating manual - OmniCore*.

4.3.2.1 How to get started with a coordinated track motion
Continued

	Action	Info/illustration
7	Activate the track unit in the jogging window and check that the coordination is working satisfactorily. This may be done by choosing **World** or **Work Object** in the field **Coordinate System** and then jogging the track axis. The robot TCP should not move, but be fixed relative to the object coordinate system.	

ℹ️ **Note**

If the robot base frame is rotated after the calibration of the track base frame, a new base frame calibration of the robot has to be done and also a new baseframe calibration of the track.

4.3.2.2 Defining the base frame for a track motion

Prerequisites

To make coordinated track motion possible it is necessary to define the base frame of the track. This frame is located in the calibration position of the track (see illustration below).

For the definition of a track base frame you need a world fixed tip within the robot's working range. The calibration procedure consists of a number of positionings of the TCP to the reference point (world fixed tip).

Please note that before the base frame of the track may be defined, the base frame of the robot must be defined with the track in the calibration position, that is robot base frame identical with track base frame.
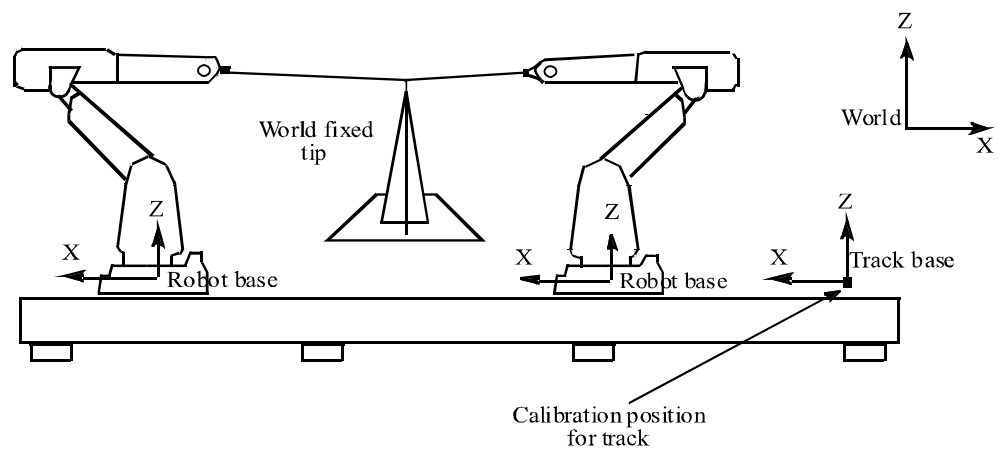
en0600002745

Definitions for track base coordinate system

The track's base coordinate system has its origin in the robot's base when the track is in its calibration position. The x direction is pointing along the linear track path and the **z** axis of the track's coordinate system is parallel with the **z** axis of the robot's base coordinate system.

The illustration below shows an example of how the base systems are oriented for a specific robot mounting. In this case the robot is mounted on the track at an angle of 45 degrees.
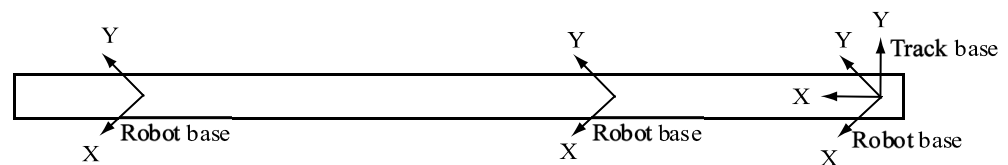
en0600002757

Base frame definition procedure

	Action
1	Tap **Calibrate**.

Continues on next page

4 Configuration

	Action
2	Select the mechanical unit for the track.
3	In the **Calibration Methods** menu, select **Base Frame** and **3 points**.
4	Activate the track unit and run it to the calibration position, that is zero position should be displayed on the FlexPendant.
5	Select **Point1**.
6	Jog the robot as close as possible to the world fixed tip.
7	Modify the position by tapping **Modify Position**.
8	Move the robot along the track and repeat the steps above for the points **Point 2** and **Point 3**.
9	Press **OK** to calculate the base frame for the track.

Result

The result of the calculation is displayed (expressed in the world coordinate system). The following values are shown:

Listed values	Description
Method	Displays the selected calibration method.
Max error	The maximum error for one positioning.
Min error	The minimum error for one positioning.
Mean error	The accuracy of the robot positioning against the tip.
Cartesian X	The x coordinate for the base frame. (x, y, z is the same as for the robot base frame).
Cartesian Y	The y coordinate for the base frame.
Cartesian Z	The z coordinate for the base frame.
Quaternion 1-4	Orientation components for the base frame.

If the estimated error is acceptable, press **OK** to confirm the new track base frame.

If the estimated error is unacceptable, press **Cancel** to redefine the calibration.

4.3.3 Coordinated positioners

4.3.3.1 How to get started with a coordinated (moveable) user coordinate system

Coordination procedure

In the checklist below, the steps required to coordinate a user coordinate system are described. In each step, there may be a reference to another chapter in this manual, where more details of the specific actions to be taken will be found.

	Action	Information
1	Calibrate the robot and the positioner, i.e. the zero position of the measuring system for both robot and positioner must be carefully determined.	See section *Calibrating* in *Operating manual - OmniCore*.
2	Define the base frame of the robot.	See *4 points XZ calibration* in *Operating manual - OmniCore*.
3	Define the user frame of the positioner.	See *Defining the user frame for a rotational single axis on page 48* or *Defining the user frame for a multi axes positioner on page 51*.
4	Create a backup of the system by tapping **Settings** - **Backup & Restore** - **Backup**.	See *Back up and restore systems* in *Operating manual - OmniCore*.
5	Create a new work object data and give it a name, e.g. wobj_turntable. In this work object, change the component `ufprog` to FALSE, indicating that the user object should be connected to a moveable mechanical unit. Also change the component `ufmec` to the name of the positioner (e.g. STN_1).	See section *Creating a work object* in *Operating manual - OmniCore*.
6	If you want the object frame to be displaced relative to the user frame, you may write the displacement in the x, y, z values of the `oframe` component of the work object.	For more information about the object frame, see *Operating manual - OmniCore*, section *What is a work object* and section *Defining the work object coordinate system*.
7	Activate the positioner in the jogging window and check that the coordination is working satisfactorily. This can be done by: • selecting **Work Object** in the field **Coordinate system** • selecting your work object, e.g. wobj_turntable, in the field **Work object** When jogging the positioner, the robot TCP should also move, following the moveable object coordinate system.	

💡 **Tip**

When programming, it is important to have the coordinated work object, in this case wobj_turntable, programmed as an argument in each move instruction. This will be automatically added to the move instruction, if the work object is activated in the jogging window before starting the programming.

4.3.3.2 Defining the user frame for a rotational single axis

About defining the user frame

This method will define the location of the user coordinate system of a rotational single axis positioner, relative to the world coordinate system. As it is a single axis the base frame and user frame will coincide. This user coordinate system should be used when a coordinated work object is used.

Prerequisites

The definition of a user frame for a rotational additional axis requires that the turntable on the additional axis has a marked reference point. The calibration procedure consists of a number of positionings for the robot's TCP on the reference point when the turntable is rotated to different angles (see illustration below).

en0600002760
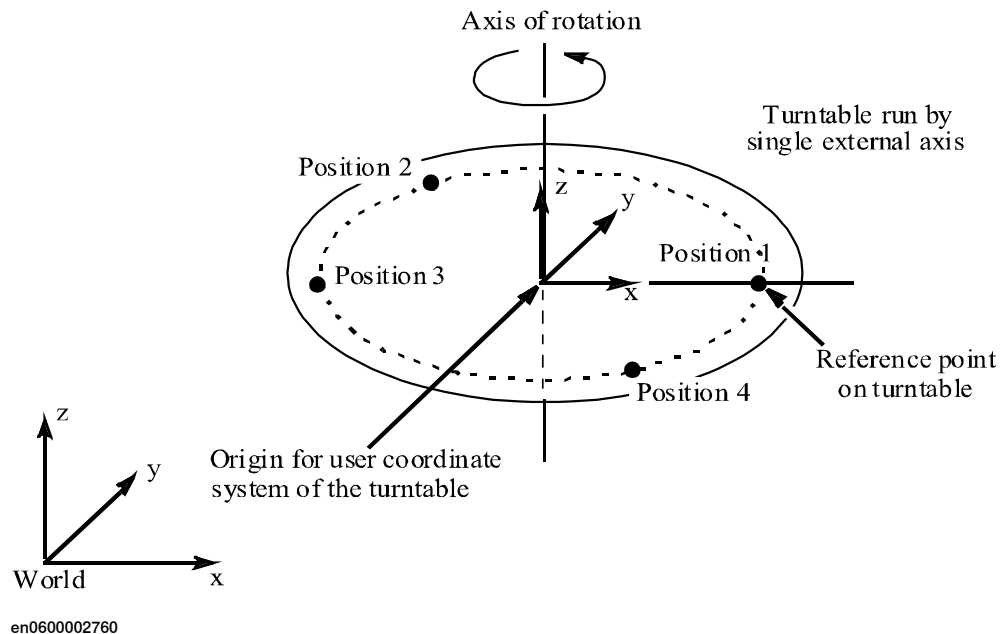
Position and directions of the user frame

The user coordinate system for the rotational axis has its origin in the centre of the turntable. The z direction coincides with the axis of rotation and the x axis goes through the reference point. The illustration below shows the user coordinate system for two different positionings of the turntable (turntable seen from above).

Continues on next page
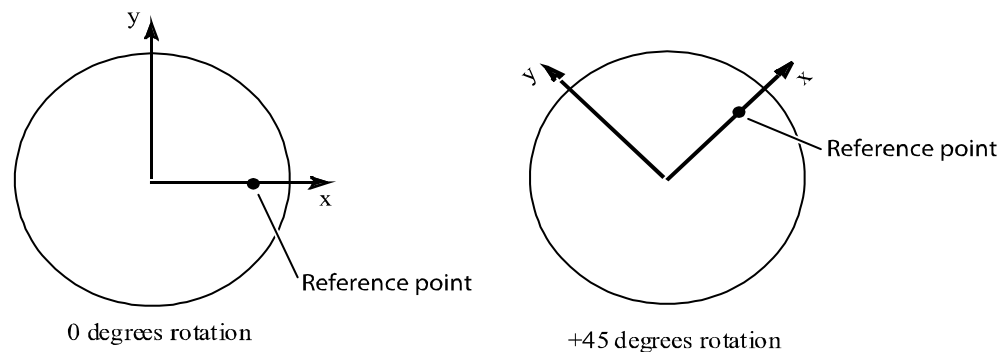
4.3.3.2 Defining the user frame for a rotational single axis
Continued

0 degrees rotation

+45 degrees rotation

en0600002761

> **ℹ️ Note**
>
> If it is intended to simulate the positioner in RobotStudio, it is recommended to define the user coordinate system of the rotational axis so that it coincides with the user coordinate system of the RobotStudio model.

User frame definition procedure

	Action
1	Tap **Calibrate**.
2	Select the rotational single axis positioner.
3	Select **Base Frame**.
4	Select the method **4 Point Z**.
5	If you have a MultiMove system, select which robot to use for the calibration. For single-robot, go directly to the next step.
6	Select **Point 1**.
7	Jog the robot as close as possible to the reference point.
8	Modify the position by tapping **Modify Position**.
9	Move the rotational positioner to a new positive position and point out the new position with the robot.
10	Repeat the steps 6-9 for the points **Point 2**, **Point 3** and **Point 4**.
11	Jog the robot to a position where the tool center point (TCP) touches an imaginary extension of the desired positive **z** axis. In this case, this point should be along the rotational axis of the turntable (above the turntable). This is only to define the positive direction of the **z** axis. It is not used to increase the accuracy of the calibration. The exact direction of the **z** axis is defined as the normal of the xy plane.
12	Select **Elongator Point Z** and tap **Modify Position**.
13	If you want, you can save the entered calibration data to a file. Tap **Positions** and then **Save**. Enter the name of the file and then tap **OK**. To restore this calibration, the file can be loaded from **Positions - Load**, instead of performing steps 6-12.
14	Press **OK** to calculate the user frame for the positioner.

49

Result

The result of the calculation is displayed (expressed in the world coordinate system). The following values are shown:

Listed values	Description
Method	Displays the selected calibration method.
Max error	The maximum error for one positioning.
Min error	The minimum error for one positioning.
Mean error	The accuracy of the robot positioning against the tip.
Cartesian X - Z	The x, y, z coordinates for the user frame.
Quarternion 1-4	Orientation components for the user frame.

If the estimated error is acceptable, press **OK** to confirm the new user frame.

If the estimated error is unacceptable, press **Cancel** to redefine the calibration.

4.3.3.3 Defining the user frame for a multi axes positioner

Parameter file required

It is possible to define positioners with more than one axis. To achieve the best possible performance from such a positioner, a set of data, describing its kinematic and dynamic properties (among other things), must be defined. This data cannot be defined in the system parameters via the FlexPendant or RobotStudio, but must be read from a parameter file. If no file was supplied with the manipulator, the manipulator cannot be coordinated with the robot. It can, however, be defined as a number of separate external axes.

Differences between one and multi axes positioner

The principles for defining a user frame for a multi axes positioner are the same as for a one axis positioner, see *Defining the user frame for a rotational single axis on page 48*. However, note that the axis must be moved in positive direction (see step 9 below).

For a positioner with more than one axis, a 4 point calibration is performed for each axis.

> ℹ️ **Note**
>
> If it is intended to simulate the positioner in RobotStudio, it is recommended to define the user coordinate systems of the rotational axes so that they coincide with the user coordinate systems of the RobotStudio model.

Define number of axes

The number of axes belonging to the positioner must be defined in the configuration file before coordinated motions are possible. The value should represent the number of axes connected in serial.

Parameter	Type	Description
no_of_error_model_joints	ROBOT_TYPE	Number of axes connected in serial belonging to the positioner. The maximum value is 6.

The parameter does not need to be set for ABB positioners.

User frame definition procedure

	Action
1	Tap **Calibrate**.
2	Select the multi axes positioner
3	Select **Base Frame**.
4	Select **4 Points for Axis 1**.
5	If you have a MultiMove system, select which robot to use for the calibration. For single-robot, go directly to the next step.
6	Select **Point 1**.

Continues on next page

4.3.3.3 Defining the user frame for a multi axes positioner
Continued

	Action
7	Jog the robot as close as possible to the reference point.
8	Modify the position by tapping **Modify Position**.
9	Move the first axis in positive direction to a new position (according to right hand rule).
10	Repeat the steps 6-9 for the points **Point 2**, **Point 3** and **Point 4**.
11	If you want, you can save the entered calibration data to a file. Tap **Positions** and then **Save**. Enter the name of the file and then tap **OK**. To restore this calibration, the file can be loaded from **Positions - Load**, instead of performing steps 6-10.
12	Select **4 Points for Axis 2** and repeat step 5-11 for the other axes belonging to the positioner.
13	Press **OK** to calculate the user frame for the positioner.

Result

The result of the calculation is displayed (expressed in the world coordinate system). The following values are shown:

Listed values	Description
Method	Displays the selected calibration method.
Max error	The maximum error for one positioning.
Min error	The minimum error for one positioning.
Mean error	The accuracy of the robot positioning against the tip.
Cartesian X - Z	The x, y, z coordinates for the user frame.
Quarternion 1-4	Orientation components for the user frame.

If the estimated error is acceptable, press **OK** to confirm the new user frame.

If the estimated error is unacceptable, press **Cancel** to redefine the calibration.

ℹ️ **Note**

When defining a work object for a coordinated motion, the user frame part of the work object is left empty (unit frame). Instead the user part is computed when needed using the kinematic model and the joint position for the mechanical unit.

---

5 Commutation

Overview of commutation

There are two methods to find the commutation value for motors that are not included in the ABB offer.

1 Automatic method, see *Commutation with service routine on page 54*.

2 Manual method, see *Manual commutation on page 56*.

> ⚠ **CAUTION**
>
> If the motor is not properly commutated, it can rev up and break.

5.1 Commutation with service routine

Service routine for commutation

The service routine **Commutation** is used to:

- Find a commutation value for a synchronous permanent magnet motor.
- Verify that the motor phase order is correct.
- Verify that the pole pair parameter value is correct.
- Verify that the resolver connection is correct.

Commutating with the service routine

	Action	Note
1	Start the service routine **Commutation**.	See *Operating manual - OmniCore* section *Programming and testing - Running a service routine*.
2	Examine the order of the motor phase connections.	*Motor phase connection order on page 54*.
3	Examine the resolver connection.	*Resolver connections on page 55*.

4	Move the motor by using the service routine.
	  • For the pre-commutated motor: Make sure that the motor phase connections are connected to the right phase. If yes, then the existing commutation is ok to use. Do not update the commutation offset.
	  • For the non-commutated motor: Commutate the motor by updating the commutation offset.
		• *Motor phase connections on page 55*.
		• *Update commutation offset on page 55*.

5	The commutation is now finished and the motor is ready to use.	

Motor phase connection order

By stepping the motor in positive direction using the service routine, the motor shaft shall turn in counter clockwise direction, if the shaft is seen from the resolver side and clockwise from the drive shaft side.
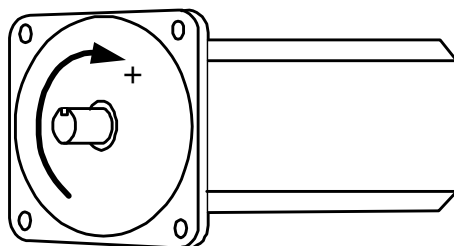
xx0400001171

If the motor is turning in the wrong direction then the motor phases have been swapped. Try changing RST to SRT, RTS, or TSR.

Continues on next page

Pole pairs parameter

Examine that the pole pairs (*Pole Pairs*) parameter is loaded with the correct value by stepping the motor from the service routine. The motor shall turn 1/16 of a revolution for every step.

Resolver connections

From the service routine step the motor in positive direction. The resolver is connected correctly if the motor angle in the jogging window is increasing. Otherwise check the wiring of the resolver.

Motor phase connections

Using the service routine, move the motor to the commutation position. For best result, commutate without any equipment connected to the motor (a free mounted motor).

There are a number of correct commutation angles (same as *Pole Pairs* parameter). If the difference between the provided commutation angle and the suggested commutation angle is a multiple of 6.283185/number of pole pairs (*Pole Pairs*)- the commutation is ok. Otherwise all motor phases shall be moved one step forward or backward (same order! RST -> STR or TRS).

A commutation value set by the motor manufacturer is normally more accurate than a value updated by the service routine.

Update commutation offset

To get a good commutation position the motor must not be affected by gravity or large friction from equipment connected to the motor. For best result, commutate without any equipment connected to the motor (a free mounted motor).

When the motor is aligned, the resolver commutation parameter can be set. When the parameter is set the database is also updated.

5.2 Manual commutation

General

Before using an additional axis, the motor must be commutated. To do the commutation manually, connect a DC power source between two nodes and then measure the position of the motor.

> **ℹ️ Note**
>
> ABB motors are pre-commutated with the commutation value 1.5708. Therefore, an ABB motor does not require modifying the commutation offset.

Prerequisites

The motor must comply with the specifications in *Motors on page 108*. The resolver must comply with the specifications in *Resolvers on page 114*.

Required material

This is a list of what you need to perform the commutation manually:

Material	Description
PC with TuneMaster	See *Application manual - TuneMaster*.
Power supply	24 V (DC). The power supply should be equipped with a relay that trips at short circuit. Otherwise a fuse will burn every time the power is applied. Check the motor data to see the current required from the power supply.
2 cable sets	Cables to brake release and motor phase. Each cable set includes one plus and one minus cable.
Motor documentation	Motor data sheet and electrical connection drawing.

Measuring the commutation position

This procedure describes how to measure the commutation position of a motor.

	Action
1	Deactivate the axis for the motor to commutate.
2	Switch off the controller.
3	Disconnect the motor cable from the motor.
4	Disconnect the motor from the gear (or in some other way make sure the motor is not affected by external torque and friction).
5	If the motor is using a brake, release it by connecting the power supply to the contact pins for the brake release. See the motor specification for maximum brake current, which contact is for the brake release, and the polarity of the contacts (if any).
6	Make sure that the brake is released by manually turning the motor.

Continues on next page

	Action
7	Connect the power supply with the plus cable to the phase S (V) and the minus cable (0 V) to the to the phase T (W).
	A short pulse is enough to move the motor to its commutation position. Disconnect the power after the voltage pulse.
8	Connect the power to give another voltage pulse to the motor. If the motor is already in its commutation position it should not move this time.
9	Disconnect the power supply from the brake release, so that the motor brake is engaged.
10	Reconnect the motor cable from the drive unit to the motor.
11	Start the controller.
12	Activate the axis.
	Do not move any mechanical unit.
13	Configure TuneMaster, selecting mechanical unit and the signal *ResolverAngle* (signal number 1).
	Zoom in on the signal to read at least 2 decimals.
	Note that the number of commutation positions are equal to the number of pole pairs. For example, a motor with 2 pole pairs have 2 possible values for this measurement. It does not matter which of the commutation points are measured.
14	Set the measured value to the parameter *Commutator Offset* in the type *Motor Calibration*.
	Restart the controller.
15	Reconnect the motor to the gear.

This page is intentionally left blank

6 Tuning

6.1 Tuning of servo control parameters

Overview

> The servo control parameters can be adjusted (tuned) to achieve the best possible motion performance.

Tuning with TuneMaster

> The recommended way to perform the tuning is by using the software TuneMaster. How to perform the tuning is described in *Application manual - TuneMaster*.

6.2 Additional tuning

6.2.1 Tuning of the soft servo parameters

General

This section details how to tune the following parameters in the type *Lag Control Master 0*:

- *K Soft Min Factor*
- *K Soft Max Factor*
- *Kp/Kv Ratio Factor*
- *Ramp Time*

> 💡 **Tip**
>
> In most applications these parameters do not have to be trimmed and can be left at their default values.

Tuning of K Soft Min Factor

The procedure below details how to make the initial tuning of the parameter *K Soft Min Factor*.

> 💡 **Tip**
>
> The movements in this trim procedure should be done close to the point where the soft servo is activated, to minimize the risk of an axis collapsing.

	Action
1	Determine a maximum axis movement for which the axis should not move, when the softness is 100%. Such a movement can be 0.1 rad for a rotating axis.
2	Determine a minimum axis movement for which the axis should move, when the softness is 100%. Such a movement can be 0.2 rad for a rotating axis.
3	Activate the soft servo with softness 100% and perform the two movements.
4	If the axis moves for both movements, the axis is too stiff and *K Soft Min Factor* should be reduced. If the axis does not move for any movement, the axis is too soft and *K Soft Min Factor* should be increased.
5	Repeat step 3 and 4 until the axis does not move for the smaller movement but does move for the bigger movement.

Tuning of K Soft Max Factor

In most cases, *K Soft Max Factor* can be left at its default value (1.0).

If the axis is too stiff at 0% softness, reduce *K Soft Max Factor*. If the axis is too soft at 0% softness, increase *K Soft Max Factor*. The tuning can be made in a similar way as for *K Soft Min Factor*, but with smaller movements.

Kp/Kv Ratio Factor

Kp/Kv Ratio Factor determines the stability margin for the axis. A value less than 1.0 increases the stability. It is not possible to set this parameter to a value larger than 1.0 since the stability of the axis would be jeopardized.

Ramp Time

If *Ramp Time* is changed, the duration of the activation and deactivation phase will change. A short ramp time can result in a twitch of the axis at activation.

6.2.2 Additional tuning for servo guns

Description in separate manual

The specifics for tuning a servo gun are described in *Application manual - Servo Gun Setup*.

7 Error handling

Handling errors - an iterative process

This section details how to handle fault localization after having performed system configuration.

Fault localization is an iterative process that must go on until all errors are eliminated. The following steps describe the order of the process:

1 Open the event log and select the category **Configuration**. Search for error messages and identify the first error.

2 Open the **Configuration Editor** in RobotStudio and correct the parameter value for the error that was found.

> **Note**
>
> To avoid new error messages, only correct one error at a time.

3 Restart the controller.

4 Go back to step 1 and proceed with the steps above until all errors are eliminated.

> **Tip**
>
> Troubleshooting is also described in the following manuals:
>
> • *Operating manual - Integrator's guide OmniCore*
> • Product manual for the robot controller
>
> All event log messages can also be found in *Technical reference manual - Event logs for RobotWare 7*

Summary

Every time a new system configuration is done or axes are tuned, fault localization and error correction must also be done to ensure that any remaining errors will be eliminated.

This page is intentionally left blank

8 System parameters

8.1 Acceleration Data

General

These parameters are applicable to each arm of the external robot in question.

Parameter description

The parameters belong to the configuration type *Acceleration Data* in the *Motion* topic.

Cfg name	Parameter name	Description
name	Name	Name of the *Acceleration Data* group. Max 32 characters.
wc_acc	Nominal Acceleration	Axis acceleration in rad/s^2 (or m/s^2 for linear axes). If the value is too high, the motor will reach the torque limit and result in poor path performance.
wc_dec	Nominal Deceleration	Axis deceleration in rad/s^2 (or m/s^2 for linear axes). If the value is too high, the motor will reach the torque limit and the axis will overshoot in fine points.

8.2 Arm

General

These parameters are applicable to each arm of the robot in question.

Parameter description

The parameters belong to the configuration type *Arm* in the *Motion* topic.

Common parameters

Cfg name	Parameter name	Description
upper_joint_bound	Upper Joint Bound	Upper bound for the axis work area (in radians or meters). The axis cannot be moved beyond this limit during jogging or program execution.
lower_joint_bound	Lower Joint Bound	Lower bound for the axis work area (in radians or meters). The axis cannot be moved beyond this limit during jogging or program execution.

Parameters for additional axes

Cfg name	Parameter name	Description
independent_joint_on	Independent Joint	Set parameter to value On in order to activate the possibility to use independent joint instructions. Default value is Off.
ind_upper_joint_bound	Independent Upper Joint Bound	Upper bound for the axis work area when operating in independent mode (in radians or meters.
ind_lower_joint_bound	Independent Lower Joint Bound	Lower bound for the axis work area when operating in independent mode (in radians or meters).

Parameters for non ABB robots

Cfg name	Parameter name	Description
name	Name	Name of the ARM data group, e.g. x.
use_arm_type	Use arm type	ID name for ARM_TYPE data group.
use_acc_data	Use acc data	ID name for ACC_DATA data group.
use_arm_calib	Use arm calib	ID name for ARM_CALIB data group.
lower_joint_bound_min	-	Minimum value for lower_joint_bound. The unit is radian or meters.
upper_joint_bound_max	-	Maximum value for upper_joint_bound. The unit is radian or meters.
cal_position	Calibration position	Calibration position. The unit is in radians or meters.

8.3 Arm Calib

General

These parameters are applicable to each arm of the external robot in question.

Parameter description

The following parameters belong to the topic *Motion* and the type *Arm Calib*.

Parameters for non ABB robots

Cfg name	Parameter name	Description
`name`	Name	Name of the ARM_CALIB data group.

8.4 Arm Type

General

These parameters are applicable to each arm of the robot in question.

Parameter description

The parameters below are the parameters used to describe a kinematic model in GEN_KIN and they are based on the standard Denavit-Hartenberg convention and follows the description provided in *Introduction to Robotics*, *Mechanics and Control* by John J. Craig. The parameters are provided for each joint and to be able to set up a kinematic model, it is necessary to understand how coordinate systems in the respective joints are transformed based on the parameters. Here coordinate system 0 is a fixed coordinate system with a certain relation to the world coordinate system. Coordinate system 1 is attached to joint 1, coordinate system 2 is attached to joint 2, etc. The description below is assumed to be provided for joint k (coordinate system k), the previous joint coordinate system is referred to as coordinate system k-1 and the next coordinate system hence k+1.

Parameters for non ABB robots

The following parameters belong to the topic *Motion* and the type *Arm Type*.

Cfg name	Parameter name	Description
name	Name	Name of the ARM_TYPE data group.
length	-	Arm length, measured along the x direction in the current coordinate system k (according to Craig's definition[i]). (Meter)
offset_z	-	Offset in z direction in current coordinate system k. Home position of a linear axis. (Meter)
theta_home_position	-	Axis angle of arm in home position, rotation around the z-axis in current coordinate system k (theta according to Craig's definition[i]). (Radians)
attitude	-	Angle between z-axis in previous coordinate system k-1 and the current coordinate system k (alpha according to Craig's definition[i]). Rotation is around the x-axis in coordinate system k-1. (Radians)

[i] The Denavit-Hartenberg notation according to John J. Craig in *Introduction to Robotics*, *Mechanics and Control*

Example 1 – XYZC(Z), available as template

Build a gantry with linear axes along X, Y and Z and a rotation around the Z axis.

In the *MOC.cfg* file, the following information will be included. See also the template files where additional info is also shown.

```
ROBOT:

-name "ROB_11" -use_robot_type "ROB11_XYZC(Z)" \
```

Continues on next page

```
-use_robot_calib "r11_uncalib" \
-use_joint_0 "rob11_1" -use_joint_1 "rob11_2" \
-use_joint_2 "rob11_3" -use_joint_3

ROBOT_TYPE:

-name "ROB11_XYC(Z)" -type "GEN_KIN3" -error_model "NOMINAL" \
-no_of_joints 4 -master_robot -tcp_robot \
-base_pose_rot_u0 0.70710678 -base_pose_rot_u1 0 \
-base_pose_rot_u2 0.70710678 -base_pose_rot_u3 0
```

The `base_pose_rot` parameters are used to get the coordinate system for the first joint such that the **z-axis is aligned with the linear movement direction, the x-axis in the world coordinate system. In addition,** `JOINT` **and** `ARM` **sections has to be set up in the** *MOC.cfg* **to be complete, see the template files. The property to decide whether the axis is rotating or linear is part of the type** `TRANSMISSION`. **By using** `rotating_move`, **the axis will be rotating around the local z-axis, otherwise the motion will be translational along the z-axis.**

With the parameters above, the type `ARM_TYPE` is shown below.

```
ARM_TYPE:

-name "ROB11_1" -length 0 -offset_z 0 -attitude 0 \
-theta_home_position 0

-name "ROB11_2" -length 0 -offset_z 0 -attitude 1.5707963 \
-theta_home_position 1.5707963

-name "ROB11_3" -length 0 -offset_z 0 -attitude 1.5707963 \
-theta_home_position 0

-name "ROB11_4" -length 0 -offset_z 0 -attitude 0 \
-theta_home_position 0
```


World coordinate system   Coordinate system 0 or 1   Coordinate system 2   Coordinate system 3 or 4
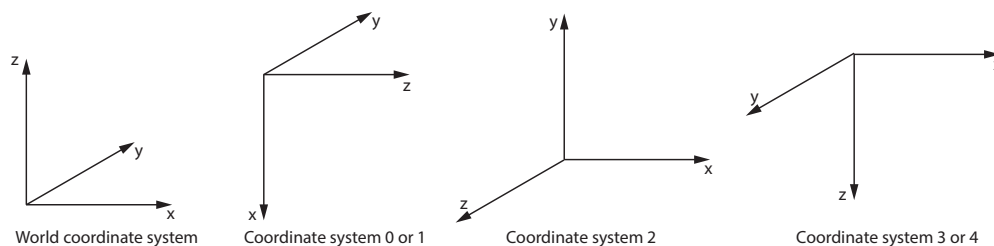
xx1800001533

1   Transformation from world coordinate system to the joint 1 coordinate system is done by the `base_pose_rot` parameters in `ROBOT_TYPE`. **Joint 1 is linear and moves along the z-axis of coordinate system 1, which is aligned with the x-axis of the world coordinate system.**

2   The Joint 1 coordinate system is rotated around the **x-axis of coordinate system 1 to get the z-axis of coordinate system 2 aligned with the negative**

Continues on next page

y-axis in the world coordinate system by a rotation of π/2. This is achieved by using the `attitude` parameter. To prepare for the coordinate system 3 to have x-axis aligned with world coordinate system, a rotation is also performed around the **z-axis** of the coordinate system 2 by setting the `theta_home_position` **parameter to π/2.**

3 To achieve a final coordinate system with negative **z-axis** aligned with the world coordinate system, the `attitude` parameter is used, providing a rotation around the **x-axis** in coordinate system 2 by π/2 to achieve coordinate system 3 and 4.

Additional rotations, such as a spherical wrist or an additional linear axis can be attached using similar steps as above.

8.5 Brake

General

These parameters control the emergency brake behavior. They are applicable to each additional axis with a brake strong enough to hold against gravitation.

Parameter description

The parameters belong to the configuration type *Brake* in the *Motion* topic.

Cfg name	Parameter name	Description
name	Name	ID name of the brake.
control_off_delay_time	Control Off Delay	The motor torque is turned off after this delay time has expired. It must be long enough to ensure that the mechanical brake has started working, or the axis risk dropping toward the ground.
brake_control_on_delay_time	Brake Control On Delay	Period of time during which deceleration by motor is used. It should be set close or equal to the mechanical brake activation time and must be long enough to damp mechanical oscillation. If the axis is still moving after this time has expired, the electrical torque brake is activated.
brake_control_on_min_delay_time	Brake Control Min Delay	Used by the brake algorithm much the same as *Control Off Delay*. Should be set to the same value as that parameter.
absolute_brake_torque	Absolute Brake Torque	Specifies max brake torque generated by the motor in the electrical torque brake phase. *Absolute Brake Torque* together with torque generated by the mechanical brake must not exceed max allowed torque for the arm, in order not to damage arm and gearbox.
brake_ramp_speed_limit	Brake Ramp Speed Limit	Specifies the speed limit for torque reduction in the electrical torque brake phase and is typically set to zero.

8.6 Drive Module User Data

General

Drive Module User Data is used to define functions for the controller cabinet.

Parameter description

The following table contains the parameters that belongs to the topic *Motion* and the type *Drive Module User Data*.

Cfg	Parameter name	Description
enable_addition-al_axes_ptc_supervi-sion	Enable additional axes ptc supervision	Enables the PTC supervision on additional axes. The default value is FALSE.

8.7 Force Master

General

Force Master is used to define how a servo gun behaves during force control. The parameters only affect the servo gun when it is in force control mode.

Parameter description

The following table contains the parameters that belongs to the topic *Motion* and the type *Force master*.

Parameters for additional axes

Cfg	Parameter name	Description
bandwidth_ramping	References Bandwidth	The frequency limit for the low pass filter for reference values.
ramp_time_switch	Use Ramp Time	Determines if the ramping of the tip force should use a constant time or a constant gradient.
ramp_torque_ref_closing	Ramp when Increase Force	Determines how fast force is built up while closing the tool when *Use ramp time* is set to No.
ramp_time	Ramp Time	Determines how fast force is built up while closing the tool when *Use Ramp Time* is set to Yes.
bandwidth_lp	Collision LP Bandwidth	Frequency limit for the low pass filter used for tip wear calibration.
alarm_torque	Collision Alarm Torque	Determines how hard the tool tips will be pressed together during the first gun closing of new tips calibrations and tool change calibrations.
col_speed	Collision Speed	Determines the servo gun speed (m/s) during the first gun closing of new tips calibrations and tool change calibrations.
distance_to_contact_position	Collision Delta Position	Defines the distance the servo tool has gone beyond the contact position when the motor torque has reached the value specified in *Collision Alarm Torque*.
force_ready_detection_bandwidth	Force Detection Bandwidth	The feedback motor speed is filtered through a LP filter with this bandwidth. This is to avoid that variations in the speed will trigger the force detection too early.
force_ready_detection_speed	Force Detection Speed	When the feedback motor speed is below this value, it is considered that the ordered force is reached.
delay_ramp	Delay Ramp	Delays the starting of torque ramp when force control is started.

8.8 Force Master Control

General

These parameters are used to set the speed limit and speed loop gain as functions of the torque.

Parameter description

The following parameters belong to the topic *Motion* and the type *Force Master Control*.

Cfg name	Parameter name	Description
no_of_posts	No. of Speed Limits	The number of points used to define speed limit and speed loop gain as functions of the torque. Up to 6 points can be defined.
torque_1 - torque_6	Torque 1- Torque 6	The torque levels, corresponding to the ordered tip force, for which the speed limit and speed loop gain values are defined.
speed_lim_1 - speed_lim_6	Speed Limit 1-6	*Speed Limit 1* to *Speed Limit 6* are used to define the maximum speed depending on the ordered tip force.
Kv_1 - Kv_6	Kv 1-6	*Kv 1* to *Kv 6* are used to define the speed loop gain for reducing the speed when the speed limit is exceeded.

8.9 Joint

General

These parameters are used to identify individual axes.

Parameter description

The following parameters belong to the topic *Motion* and the type *Joint.*

Parameters for additional axes

Cfg name	Parameter name	Description
logical_axis	Logical Axis	Used by RAPID programs to identify individual axes. Robots from ABB normally use the values 1-6, while additional axes use 7-12. E.g. the value 7 of *Logical Axis* corresponds to eax_a in the data type robtarget, 8 corresponds to eax_b, etc.

8.10 Lag Control Master 0

General

The type *Lag Control Master 0* is normally used for regulation of axes without any dynamic model.

Parameter description

The following table contains the parameters that belong to the type *Lag Control Master 0* in the topic *Motion*.

Common parameters

Cfg name	Parameter name	Description
Kp	Kp, Gain Position Loop	The amplification of the position control, e.g. 15. A high value will give a stiff axis that quickly assumes its new position. The value should be large without inducing overshoot in the position or oscillations of the axis.
Kv	Kv, Gain speed loop	The amplification of the velocity control, e.g. 2. A high value gives better high frequency stiffness, better response speed and low overshoot. If the value is too high the axis will vibrate.
Ti	Ti, Integration Time Speed Loop	Integration time in the speed regulation loop. The lower the value of *Ti Integration Time Speed Loop*, the better tracking and disturbance rejection. Too low value may case oscillation or noise.
ffw_mode	FFW Mode	Feed Forward mode. Possible options are: • 0 (No): The controller is driven by the position error (lag). • 1 (Spd): The controller receives information about the desired speed of the axis. Speed is the recommended configuration. • 2 (Trq): The controller uses the desired speed and acceleration of the axis to calculate the desired motor torque. This requires knowledge of the mass inertia of the axis, which must be supplied by the user. For this reason its more difficult to tune and is only recommended for experienced users.

Parameters for additional axes

Cfg name	Parameter name	Description
use_inpos_forced_control	Forced Control Active	Determines whether forced gain control is active for this joint. If set to Yes, *Affects Forced Control* in *Supervision* should normally also be set to Yes for this joint (see below).
Kp_forced_factor	Forced Factor for Kp	The forced factor for Kp, if forced gain control is active.
Ki_forced_factor	Forced Factor for Ki	The forced factor for Ki, if forced gain control is active.
Kp_raise_time	Rise time for Kp	The rise time for forced Kp.
bandwidth	Bandwidth	This parameter should be left at its default value.

Cfg name	Parameter name	Description
delay_time	Delay	This parameter should be left at its default value.
resonance_frequency	Df	Dynamic factor. This parameter is only available in the Trq configuration. It can be used to damp oscillations of the axis due to mechanical resonance. Initially *Df* should be left at its default value. It can be adjusted once the other controller parameters have been fixed.
inertia	Inertia	Total mass moment of inertia at motor side.
soft_servo_K_max_factor	K Soft Max Factor	Determines the value of the product Kp*Kv when the soft servo is used with softness 0%. *K Soft Max Factor* should be in the range 0.001-1000 (default 1.0). When the soft servo is activated with 0% softness, the control parameters Kp and Kv are be tuned such that Kp*Kv = (Kp*Kv)normal**K Soft Max Factor*, where (Kp*Kv)normal is the product of Kp and Kv during normal operation.
soft_servo_K_min_factor	K Soft Min Factor	Determines the value of the product Kp*Kv when the soft servo is used with softness 100%. *K Soft Min Factor* should be in the range 0.001-1000 (default 0.01). When the soft servo is activated with 100% softness, the control parameters Kp and Kv are tuned such that Kp*Kv = (Kp*Kv)normal**K Soft Min Factor*.
soft_servo_Kp_Kv_ratio_factor	Kp/Kv Ratio Factor	Factor used to alter the Kp/Kv ratio during soft servo. *Kp/Kv Ratio Factor* should be in the range 0.001-1000 (default 1.0). In soft servo mode, Kp and Kv are tuned such that Kp/Kv = (Kp/Kv)normal**Kp/Kv Ratio Factor*.
soft_servo_t_ramp	Ramp time	Default time for activation of the soft servo. The default value is 0.5 s.

8.11 Measurement Channel

General

The type *Measurement Channel* provides parameters that are applicable to each axis of the robot in question.

Parameter description

The following table contains the parameters that belong to the type *Measurement Channel* in the topic *Motion*.

Parameters for additional axes

Cfg name	Parameter name	Description
disconnect_at_deactivate	Disconnect at Deactivate	The measurement channel for a deactivated motor can be disconnected (Yes/No).WARNING! If the axis is moved when disconnected, the position of the axis might be wrong after reconnecting, and this will not be detected by the controller. The position after reconnection will be correct if the axis is not moved, or if the movement is less than 0.5 motor revolutions. For servo guns, there is a RAPID calibration method available (the ToolChange calibration) that will adjust any positional error caused by gun movement during disconnection.

8.12 Mechanical Unit

General

The type *Mechanical Unit* provides parameters that are used to define the Mechanical Unit.

Parameter description

The following table contains the parameters that belong to the type *Mechanical Unit* in the topic *Motion*.

Common parameters

Cfg name	Parameter name	Description
name	Name	Mechanical unit name.
use_activation_relay	Use Activation Relay	ID name for activation relay.
activate_at_start_up	Activate at Start Up	*Activate at Start Up* defines if the mechanical unit should be activated at start.
deactivation_forbidden	Deactivation Forbidden	*Deactivation Forbidden* defines if the mechanical unit is allowed to be deactivated.
has_no_brake	Unit has no brake	*Unit has no brake* defines if the mechanical unit has a brake or not. The default value is *No*, that is, that the mechanical unit has a brake.

Parameters for additional axes

Cfg name	Parameter name	Description
use_connection_relay	Use Connection Relay	ID name of the relay that must be activated when the mechanical unit is activated.
allow_move_of_user_frame	Allow Move of User Frame	The unit can move a user frame, e.g. a work object.
use_single_0 - use_single_5	Use Single 1 - Use Single 6	Defines which singles are part of the mechanical unit. Corresponds to the parameter *Name* in the type *Single*.

Parameters for non ABB robots

Cfg name	Parameter name	Description
use_run_enable	Use Run Enable	ID name for run enable input signal.

8.13 Motion Planner

Parameter description

The following table contains the parameters that belong to the type *Motion Planner* in the topic *Motion*.

Parameters for non ABB robots

Cfg name	Parameter name	Description
name	Name	Motion Planner name.
brake_on_timeout	Brake on Time	Brake activation time in motor on state (in seconds). Min.=0.3, Max.=3600000, Default=3600000.
dynamic_resolution	Dynamic Resolution	(Dynamic sample time)/0.024192. Min.=0.1667, Max.=1.0, Default=1.0.
path_resolution	Path Resolution	(Geometric sample time)/0.024192. If a very low programmed speed (less than 1 mm/s) is used, a small variation of the speed can be observed. This oscillation of the speed can be reduced by increasing path_resolution.
std_servo_queue_time	Queue Time	Standard servo queue time. Min.=0.004032, Max.=0.290304, Default=0.096768.
perpendicular_acc_ratio	-	Perpendicular acceleration ratio. Min.=0.1, Max.=1.5.
dyn_ipol_decbuf_type	-	OPTIMAL_TIME - original, OPTIMAL_PATH - less high torques.
micro_ipol_type	-	micro ipol type 0,.,n. Min.=0.
cpu_load_added_to_dsp	-	Used to verify CPU load margin in DSP.Min=0, Max=25, Default=0.
motion_sup_max_level	Motion Supervision Max Level	Maximum motion sup level and tune value. Min.=10, Max.=500, Default=300.

8.14 Motion System

Parameter description

The following table contains the parameters that belong to the type *Motion System* in the topic *Motion*.

Parameters for non ABB robots

Cfg name	Parameter name	Description
name	Name	Motion system name. Min.=-100, Max.=100.
min_temp_ambient_cabinet	Min Ambient Temperature Cabinet	Minimum ambient temperature for the cabinet. Min.=-100, Max.=100.
max_temp_ambient_cabinet	Max Ambient Temperature Cabinet	Maximum ambient temperature for the cabinet. Min.=-100, Max.=100.
min_temp_ambient_robot	Min Ambient Temperature Robot	Minimum ambient temperature for the robot. Min.=-100, Max.=100.
max_temp_ambient_robot	Max Ambient Temperature Robot	Maximum ambient temperature for the robot. Min.=-100, Max.=100.

8.15 Motor

General

The type *Motor* provides parameters that are applicable to each axis of the robot in question.

Parameter description

The following table contains the parameters that belong to the type *Motor* in the topic *Motion*.

Parameters for non ABB robots

Cfg name	Parameter name	Description
name	Name	Name of the MOTOR data group.
use_motor_type	Use Motor Type	ID name of the MOTOR_TYPE data group.
use_motor_calib	Use Motor Calibration	ID name of the MOTOR_CALIB data group.
stator_cooling_factor	-	Cooling factor for the stator, multiplied with attribute torque_0. Min.=0, Max.=10, Default=1.

8.16 Motor Calibration

General

The type *Motor Calibration* provides parameters that are applicable to each axis of the robot in question

Parameter description

The following table contains the parameters that belong to the type *Motor Calibration* in the topic *Motion*.

Common parameters

Cfg name	Parameter name	Description
name	Name	Name of the MOTOR_CALIB data group.
com_offset	Commutator Offset	The motor angle when voltage is applied between the phases S and T. For ABB motors *Commutator offset* should always be 1.5708.
cal_offset	Calibration Offset	Can be updated by moving the axes to their calibration positions and then fine calibrating.

Parameters for non ABB robots

Cfg name	Parameter name	Description
valid_com_offset	Commutator Offset Valid	**Yes If** com_offset **is valid.**
valid_cal_offset	Calibration Offset Valid	**Yes if** cal_offset **is valid.**

8.17 Motor Type

General

> The type *Motor Type* provides parameters that are applicable to each axis of the robot in question.

Parameter description

> The following table contains the parameters that belong to the type *Motor Type* in the topic *Motion*.

Common parameters

Cfg name	Parameter name	Description
pole_pairs	Pole Pairs	Number of pole pairs.
ke	ke Phase to Phase	Nominal voltage constant, induced voltage phase to phase (1V/1000rpm <=> 0.00955 Vs/rad) The unit is Vs/rad.
i_max	Max Current	Max. current without irreversible demagnetization. The unit is A rms.
r_stator_20	Phase Resistance	Stator phase resistance at 20 degrees Celsius. If the resistance is measured phase-to-phase, the value should be divided by 2. The unit is ohm.
l_stator	Phase inductance	Stator phase inductance at zero current. The value should be measured at a frequency of about 120Hz to correspond to what the drive expects. If the inductance is measured phase-to-phase, the value should be divided by 2. The unit is Henry.

Parameters for non ABB robots

Cfg name	Parameter name	Description
inertia	-	Motor and resolver inertia on motor side. The unit is kgm^2.
torque_0	Stall Torque	Stall torque, infinite time, temp_stator_rise to temp_stator_max. The unit is Nm.
ke_temp_coef_20	-	Temperature reduction coefficient for ke, a t 20 degrees. The unit is 1/K.
ke_stability_coef_20	-	Long-term stability reduction constant for ke after 4000 hours.
ke_tolerance_min	-	Minimum tolerance for ke (%/100) Min. ke= ke*(1+ke_tolerance_min).
ke_tolerance_max	-	Maximum tolerance for ke (%/100). Max. ke= ke*(1+ke_tolerance_max).
ke_red_2i0	-	Current dependant reduction of ke at two times rated current (%/100).

Application manual - Additional axes
3HAC082287-001 Revision: G

Cfg name	Parameter name	Description
torque_losses_at_speed1	-	Total torque losses due to friction and iron losses at speed1 (cf.below). The unit is Nm.
torque_losses_at_speed2	-	Total torque losses due to friction and iron losses at speed2 (cf.below). The unit is Nm.
torque_losses_at_speed3	-	Total torque losses due to friction and iron losses at speed3 (cf.below). The unit is Nm.
speed1	-	The speed at which torque_losses_at_speed1 is defined in rad/s.
speed2	-	The speed at which torque_losses_at_speed2 is defined in rad/s.
speed3	-	The speed at which torque_losses_at_speed3 is defined in rad/s.
temp_stator_max	-	Maximum temperature for the stator winding. The unit is degrees Celsius.
temp_stator_rise	-	Maximum temperature rise for the stator winding. The unit is degrees Celsius.
temp_rotor_max	-	Maximum temperature for the rotor. The unit is degrees Celsius.
temp_rotor_rise	-	Maximum temperature rise for the rotor. The unit is degrees Celsius.
r_stator_temp_coef_20	-	Temperature coefficient for the stator resistance at 20 degrees Celsius.

8.18 Robot

Parameter description

The following table contains the parameters that belong to the type *Robot* in the topic *Motion*.

A maximum of 8 instances of the type *Robot* can be configured in a system.

Parameters for non-ABB robots

Cfg name	Parameter name	Description
name	Name	Name of the robot, for example master.
use_robot_type	Use Robot Type	
use_joint_0	Use Joint 1	ID name of 1st axis, for example robx_1.
use_joint_1	Use Joint 2	ID name of 2nd axis, for example robx_2.
use_joint_2	Use Joint 3	ID name of 3rd axis, for example robx_3.
use_joint_3	Use Joint 4	ID name of 4th axis, for example robx_4.
use_joint_4	Use Joint 5	ID name of 5th axis, for example robx_5.
use_joint_5	Use Joint 6	ID name of 6th axis, for example robx_6.
base_frame_pos_x	Base Frame x	Base frame position in respect of world frame coordinate system, x - direction (meters). Min.=1000, Max.=1000, Default=0.
base_frame_pos_y	Base Frame y	Base frame position in respect of world frame coordinate system, y - direction (meters).
base_frame_pos_z	Base Frame z	Base frame position in respect of world frame coordinate system, z - direction (meters).
base_frame_orient_u0	Base Frame q1	Base frame orientation in respect of world frame coordinate system, first quaternion (q1). Min.=-1, Max.=1, Default=0.
base_frame_orient_u1	Base Frame q2	Base frame orientation in respect of world frame coordinate system, second quaternion (q2).
base_frame_orient_u2	Base Frame q3	Base frame orientation in respect of world frame coordinate system, third quaternion (q3).
base_frame_orient_u3	Base Frame q4	Base frame orientation in respect of world frame coordinate system, fourth quaternion (q4).

Continues on next page

Cfg name	Parameter name	Description
base_frame_moved_by	Base Frame Moved by	*Base Frame Moved by* defines the name of robot or single that moves the base frame of the robot. ℹ️ **Note** When a robot is coordinated with 3rd party track, the performance of the track will be reduced to match limitations from the robot.
rot_x_tol	Orientation Tolerance about x	Orientation tolerance (in radians) Min.=0, Max.=4, Default=0.001.
rot_y_tol	Orientation Tolerance about y	Orientation tolerance (in radians). Min.=0, Max.=4, Default=0.001.
rot_z_tol	Orientation Tolerance about z	Orientation tolerance (in radians). Min.=0, Max.=4, Default=0.001.
robot_weight_in_power_sharing	Robote weight in power sharing	Defines how power should be distributed between robots and singles in case the power supply is a limiting factor. ℹ️ **Note** Must be defined for low power drive systems (B5, E5 and E9) and C90XT Type A with additional axes.

8.19 SG Process

General

The type *SG Process* provides parameters that are applicable for servo guns.

Parameter description

The following table contains the parameters that belong to the type *SG Process* in the topic *Motion*.

Parameters for additional axes

Cfg name	Parameter name	Description
sync_check_off	Sync Check Off	By setting this parameter to 'Yes', it will be possible to close the gun without having done a tip calibration. This is useful during the tuning procedure of a servo gun or if running an application where tip calibrations are not used. When running the gun in production, it is recommended to always have the sync check active in order to prevent possible damage caused by closing an unsynchronized pair of gun tips.
min_close_time_adjust	Close Time Adjust	Constant time adjustment (s), positive or negative, of the moment when the gun tips reach contact during a gun closure. This value is normally zero. May be used to delay the closing slightly when the synchronized pre closing is used for welding.
close_position_adjust	Close Position Adjust	When the tool tips reach the position (plate thickness) ordered by the close instruction, the force control starts. This tool tip position can be adjusted with *Close Position Adjust* to make the force control start earlier.
pre_sync_delay_time	Force Ready Delay	Constant time delay (s) before sending the weld ready signal after reaching the programmed force.
max_motor_torque	Max Force Control Motor Torque	Maximum allowed motor torque (Nm) during force control. The parameter will protect the gun from too high programmed force, by reducing the resulting motor torque to this upper level. A warning will be logged whenever this happens. The value must not be set higher than the *Torque abs. max* (type *Stress duty cycle*) which defines the maximum output of motor torque during both force and position control.
post_sync_time	Post-synchronization Time	Release time anticipation (s) of the next robot movement after a weld. This parameter can be tuned to synchronize the gun opening with the next robot movement. The synchronization may fail if the parameters is set too high.
calib_mode	Calibration Mode	The number of closings performed during a Tipwear calibration. Normally 2 closings will be ok. An increase may improve the accuracy of thickness detection for some servo guns.

Continues on next page

Cfg name	Parameter name	Description
`calib_force_high`	Calibration Force High	The maximum tip force (N) used during a Tip-Wear calibration. For best result of the thickness detection, it is recommended to use the max programmed weld force.
`calib_force_low`	Calibration Force Low	The minimum tip force (N) used during a Tip-Wear calibration. For best result of the thickness detection, it is recommended to use the minimum programmed weld force.
`calib_time`	Calibration Time	The wait time (s) during a calibration before the positional gun tip correction is done. Recommended value ca: 0.5 s.
`no_of_active_db_posts`	Number of Stored Forces	Number of stored forces in the force VS motor torque table. The minimum value allowed is 2.
`squeeze_force_1` - `squeeze_force_10`	Tip Forces 1 - 10	Gun tip force 1 (N) - Gun tip force 10 (N).
`squeeze_torque_1` - `squeeze_torque_10`	Motor Torque 1 - 10	Motor torque 1 (Nm) - Motor torque 10 (Nm).

8.20 Single

Parameter description

The following table contains parameters that belong to the topic *Motion* and the type *Single*.

A maximum of 12 instances of the type *Single* can be configured in a system.

Parameters for additional axes

Cfg name	Parameter name	Description
`name`	Name	The name of the single. A single axis mechanical unit without kinematic model must have the name of single 1 set to the same name as the mechanical unit.
`use_single_type`	Use Single Type	Defines which single type to use.
`single_weight_in_power_sharing`	Single weight in power sharing	Defines how power should be distributed between robots and singles in case the power supply is a limiting factor. ![i] **Note** Must be defined for low power drive systems (B5, E5 and E9) and C90XT Type A with additional axes.

8.21 Single Type

Parameter description

The following table contains the parameters that belong to the topic *Motion* and the type *Single Type*.

Parameters for additional axes

Cfg name	Parameter name	Description
mechanics	Mechanics	• TRACK - Linear motion. • FREE_ROT - Rotating additional axis. • SG_LIN - Linear servo gun motion.

8.22 Stress Duty Cycle

General

The type *Stress Duty Cycle* provides parameters that are applicable to each axis of the robot in question.

Parameter description

The following table contains the parameters that belong to the type *Stress Duty Cycle* in the topic *Motion*.

Common parameters

Cfg name	Parameter name	Description
name	Name	Name of the STRESS_DUTY_CYCLE.
speed_absolute_max	Speed Absolute Max	The absolute highest motor speed to be used. (rad/s)
torque_absolute_max	Torque Absolute Max	The absolute highest motor torque to be used. (Nm) • **For non ABB robots: If** torque_absolute_max **is too high it may result in a configuration error at restart. To avoid this make sure that:** torque_absolute_max **< sqrt(3)***ke*i_max**.**

8.23 Supervision

General

> The type *Supervision* provides parameters that are applicable to each axis of the robot in question.

Parameter description

> The following table contains the parameters that belong to the type *Supervision* in the topic *Motion*.

Parameters for additional axes

Cfg name	Parameter name	Description
`joint_affect_forced_Kp`	Affects Forced Control	Determines whether this joint effects forced gain control.
`Kp_forced_on_limit`	Forced on Position Limit	The upper position limit for forced gain control.
`Kp_forced_off_limit`	Forced off Position Limit	The lower limit for forced gain control.

Parameters for non ABB robots

Cfg name	Parameter name	Description
`name`	Name	Name of the SUPERVISION data group.
`use_supervision_type`	Use Supervision type	ID name of SUPERVISION_TYPE.
`power_up_position_on`	Power Up Position Supervision	Power up position supervision On, default is Off.
`counter_supervision_on`	Counter Supervision	Counter supervision On, default is Off.
`position_supervision_on`	Position Supervision	Position supervision On, default is Off.
`speed_supervision_on`	Speed Supervision	Speed supervision On, default is Off.
`load_supervision_on`	Load Supervision	Load supervision On, default is Off.
`jam_supervision_on`	Jam Supervision	Jam supervision On, default is Off.
`in_position_range`	In Position Range	-
`normalized_zero_speed`	Zero Speed (%)	-
~~dsp_torque_limitation_zero_speed_width~~	-	Deadband speed width (in rad/s on motor side).

8.24 Supervision Type

General

The type *Supervision Type* provides parameters that are applicable to each axis of the robot in question.

Parameter description

The following table contains the parameters that belong to the data group *Supervision Type* in the topic *Motion*.

Parameters for non ABB robots

Cfg name	Parameter name	Description
name	Name	Name of the SUPERVISION_TYPE data group.
static_power_up_position_limit	-	Static power up position error limit at zero speed.The unit is radians, Min.=0 and Max.=30.
dynamic_power_up_position_limit	Dynamic Power Up Position Limit	Dynamic power up position error limit at zero speed, the unit is radians.
static_position_limit	-	Position error limit at zero speed, the unit is radians on motor side.
dynamic_postion_limit	-	Position error limit at max speed, the unit is radians on motor side.
static_normalized_speed_limit	-	Speed error limit at zero speed. (% max. speed).
dynamic_normalized_speed_limit	-	Speed error limit at max speed (% max speed).
normalized_influence_sensitivity	-	Speed error influence sensitivity reduction. (% max. speed).
speed_half_time	-	Declination factor half time for supervision limits. The unit is seconds, Min=0 and Max.=5.
max_jam_normalized_speed	-	Speed limit for jam versus overload supervision. (% max. speed).
max_overload_time	-	Maximum overload time. The unit is seconds, Min.=0 and Max.=20.
max_jam_time	Max Jam Time	Max jam time. The unit is seconds, Min.=0 and Max.=20.
teach_mode_speed_max_main	Teach Max Speed Main	Maximum ordered speed ratio in teach mode (% max speed). Min.=0, Max.=1, Deafult=0.15.
teach_mode_speed_max_dsp	Teach Max Speed DSP	Maximum supervision speed ratio in teach mode for axis computer (% max speed). Min.=0, Max.= 1, Default=0.28.

8.25 Transmission

General

The type *Transmission* provides parameters that are applicable to each arm of the robot in question.

Parameter description

The following table contains the parameters that belong to the type *Transmission* in the topic *Motion*.

Common parameters

Cfg name	Parameter name	Description
name	Name	Name off the TRANSMISSION data group.
transm_joint	Transmission Gear Ratio	Gear ratio between motor and axis. For linear axis gear ratio is specified as motor rotation in radians per meter linear move (21.43 denotes that when the motor rotates 21.43 radians - the axis moves 1 m).
rotating_move	Rotating Move	Denotes whether the axis is of the rotating type (Yes) or linear type (No).
high_gear	Transmission Gear High	The integer value of the numerator of the transmission gear ratio. Only used for independent joints.
low_gear	Transmission Gear Low	The integer value of the denominators of the transmission gear ratio. Only used for independent joints.Example: For a rotating axis with high gear 100 and low gear 30, has a transmission gear ratio of 100/30=3.333333.

8.26 Uncalibrated Control Master 0

General

The type *Uncalibrated Control Master 0* provides parameters that are applicable to each axis of the robot in question.

Parameter description

The following table contains the parameters that belong to the type *Uncalibrated Control Master 0* in the topic *Motion*.

Common parameters

Cfg name	Parameter name	Description
Kp	Kp, Gain Position Loop	The amplification of the position control, e.g. 15. A high value will give a stiff axis that quickly assumes its new position. The value should be large without inducing overshoot in the position or oscillations of the axis.
Kv	Kv, Gain Speed Loop	The amplification of the velocity control, e.g. 2. A high value gives better high frequency stiffness, better response speed and low overshoot. If the value is too high the axis will vibrate.
Ti	Ti Integration Time Speed Loop	Integration time in the speed regulation loop. The lower the value of *Ti Integration Time Speed Loop*, the better tracking and disturbance rejection. Too low value may case oscillation or noise.
speed_max_n	Speed Max Uncalibrated	Max speed for uncalibrated axis (rad/s on motor side).
acc_max_n	Acceleration Max Uncalibrated	Max acceleration for uncalibrated axis (rad/s2 on motor side) Recommended value: *Nominal Acceleration * Transmission Gear Ratio*.
dec_max_n	Deceleration Max Uncalibrated	Max deceleration for uncalibrated axis (rad/s2 on motor side) Recommended value: *Nominal Deceleration * Transmission Gear Ratio*.

9 Hardware
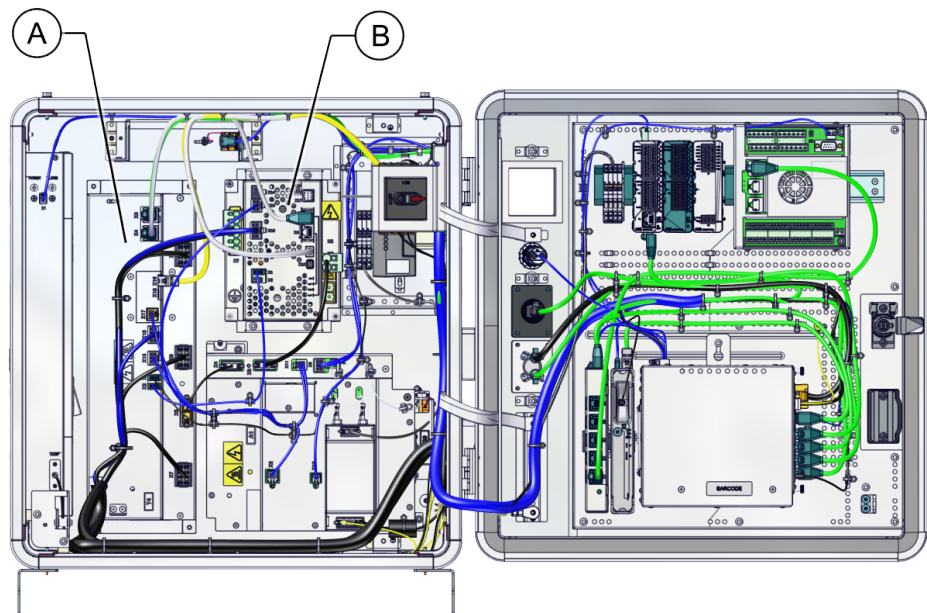
9.1 Configuration of the drive system

General

The drive system in the OmniCore controller contains one main drive unit and one or more additional drive units.

Location

The location of the drive system in the OmniCore controller is shown in the following images.

C90XT Type A

xx2400001395

A	Main drive unit
B	Additional drive unit (for additional axes)

9 Hardware

V line (V250XT and V400XT)

xx2200000703

A	Main drive unit
B	Additional drive unit (for additional axes)

For an exact location, see the product manual for the respective controller variant listed in *References on page 7*.

Units

The power unit, drive unit, and additional drive units can be placed in the following positions in the controller.

Position	Identification	Description	Art. no.	Note
1	DCQC3069	Power unit LVHP	3HAC082501-001	
	DCQC3070	Power unit HVHP	3HAC082500-001	
2	DSQC3062	Drive unit HV	3HAC082498-001	
	DSQC3084	Drive unit LV	3HAC083415-001	
3	DSQC3065	Additional drive unit	3HAC064983-001	

Continues on next page

Application manual - Additional axes
3HAC082287-001 Revision: G

Position	Identification	Description	Art. no.	Note
3-8	DSQC3065	Additional drive units	3HAC082499-001	Positions 6-7-8 are only available for V400XT.

> 💡 **Tip**
>
> For exact location and specification of how many additional drive units are allowed, see the product manual for the respective controller variant listed in *References on page 7*.
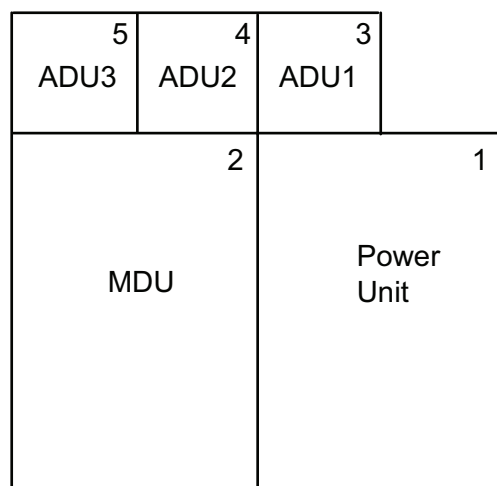
C90XT Type A

For OmniCore C90XT *Type A*, 1 additional drive unit can be installed in the following position:
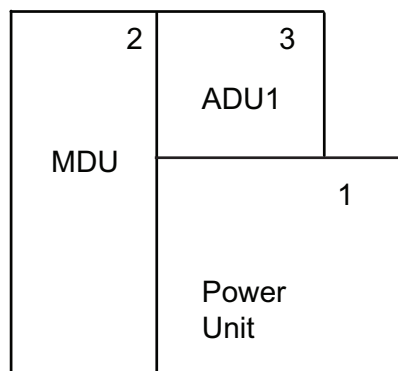
xx2400001403

V250XT Type B

For OmniCore V250XT *Type B* with high power drive systems (B4, E4 and E8), up to 3 additional drive units can be installed in the following positions:
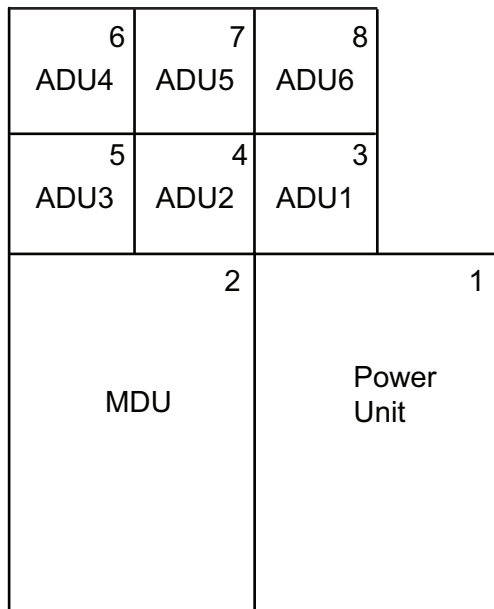

xx2300000976

9.1 Configuration of the drive system
Continued

V400XT

For OmniCore V400XT with high power drive systems (B4, E4 and E8), up to 6 additional drive units can be installed, in the following positions:

6 ADU4	7 ADU5	8 ADU6
5 ADU3	4 ADU2	3 ADU1

2 MDU	1 Power Unit

xx2200000702

V line, low power

For OmniCore V line controllers with low power drive systems (B5, E5 and E9), 1 additional drive unit can be installed in the following position:

	3 ADU1	

2 MDU	1 Power Unit
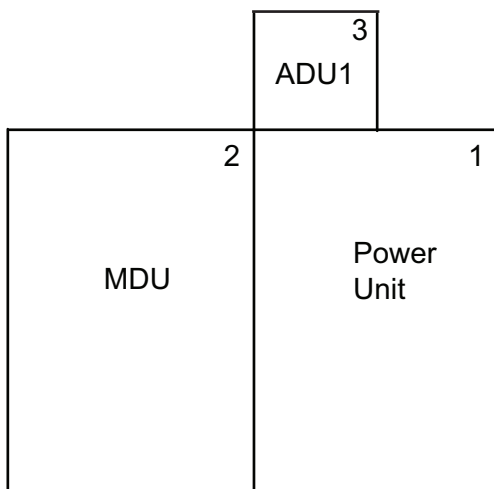
xx2400001533

ADU cables

Description	Art. no.	Note
Harn. 24V_SYS_DRV	3HAC077723-001	
Harn. Drive DC-bus	3HAC066724-001	
Harn. ADU_BRAKE	3HAC074620-001	

Continues on next page

Description	Art. no.	Note
Ethernet harness	3HAC077379-001	

9.2 Drive units

Overview

A Main Drive Unit (MDU) consists of 6 power stages.

An Additional Drive Unit (ADU) consists of one power stage.

Voltage for drive units

The following table describes the input voltage and the Dc-bus voltage for the different drive units. See also *Requirements for high voltage motors on page 108*.

Drive unit	Output voltage to motor (V_{rms}) [i]	Max dc bus voltage (V_{rms})
Drive unit, DSQC3084	nominal 230 V	430 V
Drive unit, High Voltage, DSQC3062	nominal 400 V	800 V
Additional drive unit, DSQC3065	nominal 400 V	800 V

[i] Defined as line to line.

Current limits for drive units

Drive unit	Power stage	Rated current (arms) [i]	Time limited current (arms) [ii]	Max current (arms) [iii]	Time limit for max current (s) [iv]
DSQC3065	inv_30_55	30	39	55	3

[i] The rated current is the maximum current that can be used continuously in the speed range from standstill up to max speed.
[ii] The time limited current is the maximum current that can be used at standstill during a limited time.
[iii] The max current during acceleration or deceleration during a limited time (specified by *iv*).
[iv] The max time for max current during acceleration or deceleration.

See *Example of ADU load on page 103*.

Brakes

The brake should be connected to the drive unit that controls the additional axis, on connector X15 on the drive unit.

> ⚠️ **CAUTION**
>
> If the brake is not correctly connected, then there is a risk of unexpected movement.

The brake current is displayed on the FlexPendant, in the **Settings** app under **Hardware Devices** -> **Drive Link X** -> **Runtime Information**, for drive unit devices. This data can be used for configuration and troubleshooting.

Continues on next page

Drive unit connection

The following table shows the drive unit connection for each drive unit when using configuration template files for standalone axes.

When using a template file, a power stage is connected to a physical output. The label of this output in the electrical circuit diagram is shown in the column "Designation in circuit diagram".

Drive Unit	Template file name (drive unit name)[i]	Power stage	Designation in circuit diagram	Label on unit
DSQC3062	M1 (DMX)	INV_31_54	T4	X7
	M2 (DMX)	INV_17_26		X9
	M3 (DMX)	INV_31_54		X11
	M4 (DMX)	INV_17_26		X8
	M5 (DMX)	INV_31_54		X10
	M6 (DMX)	INV_17_26		X12
DSQC3065	M7 (DMX)	INV_30_55	T41-T46	X7

[i] X= drive module number

Example of ADU load

The ADU is specified to handle 30A_RMS continuous current.

The ADU is specified to handle 39A_RMS as 3s time limited standstill current.

The ADU is specified to handle 55A_RMS as 3s time limited rotating current.

So assume we run at 55A_RMS for 3s, now we need to cool down the inverter using any of the following case in order to be able to run 55A_RMS in 3s again.

- 3.6s with 0A
- 4.6s with 14A
- 10s with 24A
- 30s with 28A

Or something in between 1-4.

9.3 Measurement system

Overview

The OmniCore C90XT Type A can control and measure up to 7 axes at the same time.

The OmniCore V250XT Type B can control and measure up to 9 axes at the same time.

The OmniCore V400XT can control and measure up to 12 axes at the same time.

Serial measurement links

Each drive unit can connect to a serial measurement link, that can be directly connected to the serial measurement board. It can also share a measurement link to another drive unit.

Serial Measurement Board

The standard SMB has seven resolver inputs. These inputs can be used as seven different nodes where the node number normally is equal to the axis number e.g. axis 1 to node 1.

Back-up battery

A back-up battery supplies the SMB with power during power failure. If an axis is moved a small distance during power off, the system is ready for operation, and no synchronization is needed after power on.
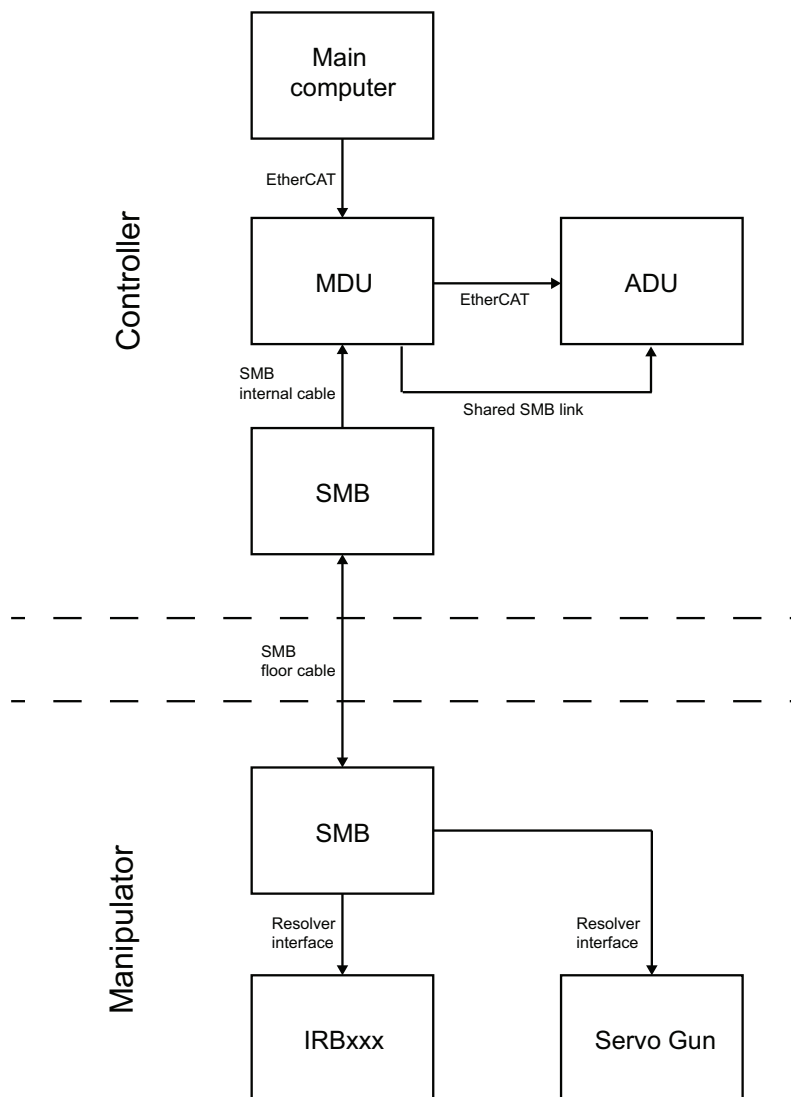
Features

Specifications for the measurement system:

- Each controller cabinet can handle two SMBs divided on two serial links.
- Each serial link can handle up to seven axes using measurement node 1-7.

9.4 Serial measurement link examples

1 ADU + drive unit

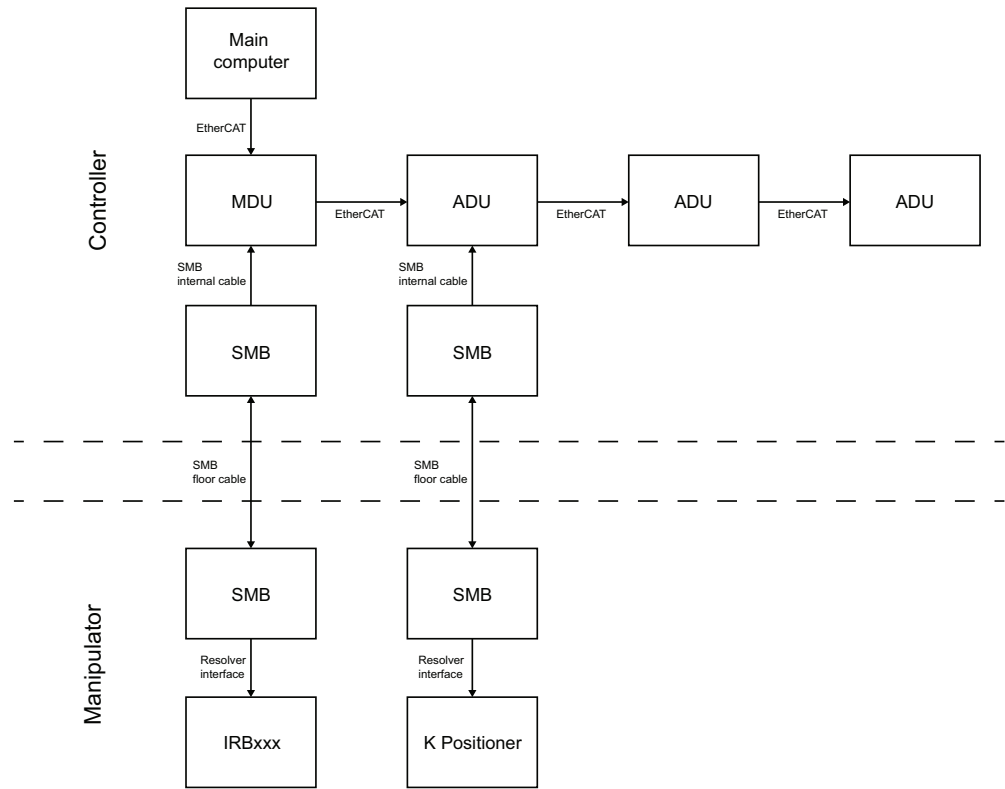The following is an example of a setup with a drive unit (MDU) and one additional drive unit (ADU):

xx2200000704

3 ADU + drive unit

The following is an example of a setup with a drive unit (MDU) and three additional drive units (ADU):

9 Hardware

xx2200000705

9.5 Equipment for additional axes

Overview

A number of parts needed to install and operate additional axes are available from ABB.

Motor units and gear units offer

The offer consists of:

- Motors
- Motors with gear boxes
- SMB boxes
- Cables

For more information, see *Product specification - Motor Units and Gear Units* and *Product manual - Motor Units and Gear Units*.

9.6 Motors

Overview

The motor units sold by ABB are specifically designed for ABB's robots and can be used for peripherals requiring power-steered motors that are synchronized with the robot movements. The motor units are designed for optimal performance and to facilitate installation and application.

> **ℹ Note**
>
> Before a motor is acquired, read also the information on how to calculate the correct motor data, see *Simple dimensioning of the motor on page 111*.

Motor description

Motor shall be a permanent magnet servo-motor of synchronous type intended for three-phase sinusoidal AC voltage, coupled in star (Y) connection.

- the motor should preferably be winded as class F according to IEC 85.
- dielectric strength minimum 1600 V. For low voltage motors connected to drive module. For high voltage motors connected to drive module, see *Requirements for high voltage motors on page 108*
- Measurement signal cables must be separated from motor cables, and cables from temperature sensor and brake.

Requirements for high voltage motors

Third party driveline components used as external equipment on the large robots (IRB 67x0 and larger) must withstand the voltage stress levels as described in the following.

These data are valid for high voltage motors connected to the drive units:

- High voltage main drive unit DSQC3062
- High voltage additional drive unit DSQC3065

The maximum allowed motor cable length is 30 m. Rise time is expressed as an indicative value at motor terminals.

Converter specifics	
Voltage (Pulse-Width Modulated)	400-480 VAC
DC link maximum voltage	790 VDC (including tolerance: 825 VDC)
Switching frequency	4 kHz
System specifics	
Rise time / dU/dt (indicative value)	0.2 microsec (as defined in IEC 60034-25) / 9 kV/microsec
Requirement for drive line components	
Insulation strength	According to IEC 60034 (i.e. >2000 V)
Voltage stress withstand capability (including PD deterioration effects)	Above withstand level B according to IEC 60034-25, Figure 17 Chapter 7

Continues on next page

Thermal protection

The temperature sensor normally used is of type PTC resistor. A high resistance or open circuit indicates that the temperature of the motor exceeds the rated level. If temperature sensor is not used, the circuit must be strapped. If more than one motor is used, all PTC resistors are connected in series.

The system input characteristics are:

- High temperature >3500 ohm
- Low temperature <3500 ohm

> ⚠️ **WARNING**
>
> The PTC supervision must be turned on in the configuration by setting the parameter *Enable additional axes ptc supervision* to *TRUE*, in the topic *Motion*, type Drive Module User Data.

> ℹ️ **Note**
>
> For F class winding with maximum temperature of 155ºC, Siemens B59135-M155_A70 can be used.

Motor connection

Positive electric rotation R ->S ->T -> (U, V, W) results in positive mechanical rotation defined as **clock wise** direction, seen from the drive shaft side. See illustration below. For connection and cabling for the motor to the controller, see the product manual for the robot controller listed in .
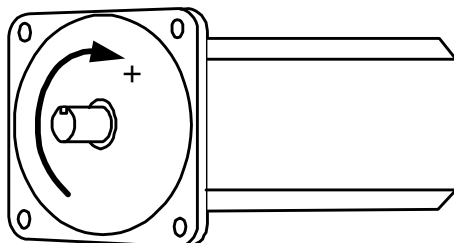
xx0400001171

Brake

Select a brake with minimum brake torque, sufficiently large to handle emergency stop when axis is moving downwards with maximum gravity. Check that maximum brake torque does not exceed allowed mechanical stress levels.

- Brake release voltage: 24 VDC +/- 10%.

> ℹ️ **Note**
>
> Check brake release voltage at maximum brake (motor) temperature and maximum allowed wear out for the brake.

Motor types

For more information about the recommended motor types from ABB, see section *Equipment for additional axes on page 107*.

9.7 Simple dimensioning of the motor

Overview

Before connecting a motor, read the general description for motors in chapter *Motors on page 108*

> **i** **Note**
>
> This section is used as a rough dimensioning of the motor, so before installing the motor make sure that it is dimensioned by a professional.

Calculate system performance

Either the motor or the drive unit sets the limitations for the system performance.

Value	Description
Kt_{min}	Motor torque constant (Nm/A_{rms}).
I_{max} drive	Max current for the drive unit (A_{rms}). See *Drive units on page 102*.
I_{max} (motor)	Max current for the motor (A_{rms}).
T_0	Average motor torque (Nm).
I_0	Average drive unit current (A_{rms}). See *Drive units on page 102*.

Calculate T_{max} and $T_{average}$ for the drive unit and the motor, then choose the limiting torque.

Criteria	Calculate the minimum value
T_{max}(system)	= min($Kt_{min}*I_{max}$(drive unit), $Kt_{min}*I_{max}$(motor))
$T_{average}$(system)	= min(T_0(motor), $Kt_{min}*I_0$(drive unit))

Check intermittence

When T_{max} and $T_{average}$ for the system is found, check the thermal load factor. It could be of importance if the additional axis accelerates slowly or if the axis moves with short quick movements without stops. The motor, or the drive unit could be over heated. Observe the planned cycle and calculate the total acceleration time. The other time is treated as static load.

T_{stat} = friction torque + gravitational torque

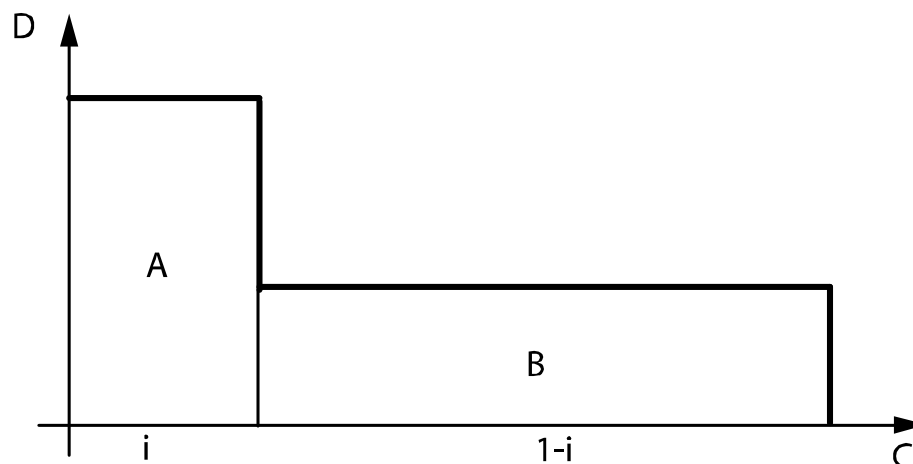| Value	Description
i	Time in acceleration and deceleration divided by total time
T_{stat}	Static load
1-i	Time in constant speed and standing still (only friction and gravity influences the motor)

Calculate: $T_{rms} = \text{sqrt}(T_{max}^2 * i + T_{stat}^2 * (1-i))$

9.7 Simple dimensioning of the motor
Continued

xx0500002231

A	Max torque (T_{max})
B	Static torque (T_{stat})
C	Time
D	Torque

Dimensioning

T_{rms} should be lower than $T_{average}$. Otherwise reduce T_{rms} or change one of the components, drive unit or motor.

Acceleration performance on arm side could now be calculated:

Acceleration = (T_{max} - GravitationalTorque - Friction) / (Inertia * Transmission)

Deceleration = (T_{max} - GravitationalTorque + Friction) / (Inertia * Transmission)

An alternative is to tune the acceleration and deceleration (parameters: *Nominal acceleration* and *Nominal deceleration*) directly on the external axis and find out if the assessable torque (T_{max}) gives desired performance.

If it is impossible to reach desired performance replace the motor or the drive unit.

Example

In this example we use worst case performance which means acceleration against the gravity

T_0	5 (Nm)
Kt_{min}	1.0 (Nm/A)
I_{max} (motor)	15 (A)
I_{max} (drive unit)	10 (A)
I_0 (drive unit)	6 (A)
intermittence	0.1
Transmission (n)	100
Mass (M)	20 (kg)
Friction (F)	2 (Nm)

Continues on next page

Application manual - Additional axes
3HAC082287-001 Revision: G

Gravity constant (g)	9.81 (N/kg)
Length to mass (L)	1.0 (meter)
Motor inertia (J_m)	0.005 (kgm^2)
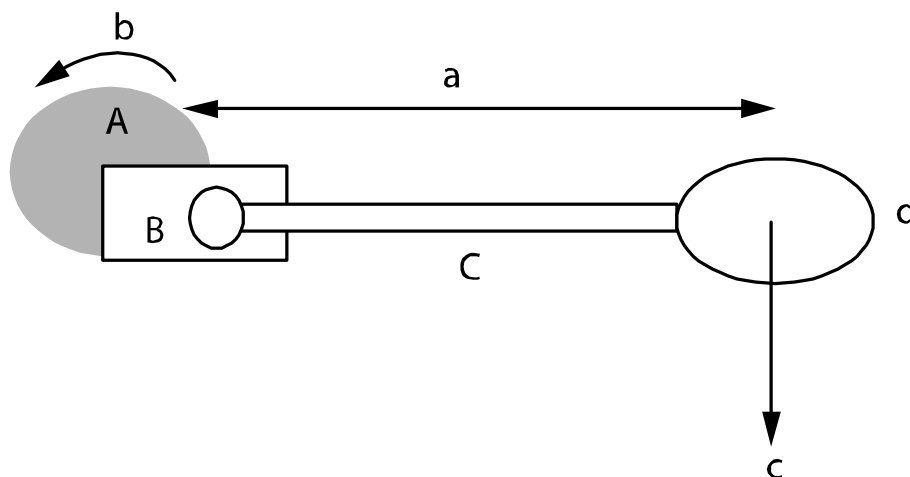
xx0500002230

a	Length to mass (L)
b	Motor inertia (J_m)
c	Mass (M) * GravityConstant (g)
d	Mass of Arm (M)
A	Motor
B	Gearbox
C	Arm

In this example acceleration needs to be 5 rad/s.

Calculations

Gravitational torque = (M*L*g)/n =(20*1*9.81)/100 = 1.96

T_{stat}= FrictionTorque + GravitationalTorque = 2 + 1.96 = 3.96

T_{max} (system) = min (Kt_{min} * I_{max} (drive unit), Kt_{min} * I_{max}(motor) = min(1*10, 1*15)=10

$T_{average}$(system) = min((T_0(motor), Kt_{min}*I_0(drive unit)) = min(5.0, 1*6) = 5.0

T_{rms}= sqrt(T_{max}^2 * i + T_{stat}^2 * (1-i)) = (10^2*0.1+3.96^2*(1-0.1) $)^{0.5}$=4.9

T_{rms} is lower than average. No need to change motor or drive unit.

Total moment of inertia on motor side J= J_m+(M*L^2)/n^2=0.005+(20*1^2)/100^2=0.007

Acceleration = (T_{max} - GravitationalTorque - Friction)/(J*n) =
(10-1.96-2)/(0.007*100)=8.6

Deceleration = T_{max} -
GravitationalTorque+Friction)/(J*n)=(10-1.96-2)/(0.007*100)=14.3

Both acceleration and deceleration are within the demand.

9.8 Resolvers

Overview

The resolver is integrated in the motors from ABB. The resolver must be approved by ABB for reliable operation.

Approved resolvers

The following resolvers are approved by ABB

Manufacturer	Article numbers
LTN Servotechnik GmbH	LTN RE-21-1-V02, size 21 LTN RE-15-1-V16, size 15
AG	V23401-U2117-C333, size 21
Tamagawa Seiki Co	TS 2640N141E172, size 21 TS 2640N871E172, size 21 TS 2620N871E172, size 15

Resolver specification

Data	Value	Unit
Single speed resolver		
Operating temperature	-25 to +120	°C
Rated input voltage	5	V_{RMS}
Frequency	4	kHz
Primary (EXC)	Rotor	
Secondary (X, Y)	Stator	
Nominal impedance - Primary (stator winding open) Z_{RO} at 4 kHz	>115	Ω
Nominal impedance - Secondary (rotor winding closed) Z_{SS} at 4 kHz	<440	Ω
Transformation ratio	0.5 ± 20%	
Phase shift out-in	0 ± 10	deg
Max error spread	≤ 10	arcmin
Resolver adjustment (COMOFF)	+90 ± 0.5	deg

The resolver has one rotor and two stator windings. The definition of the output signals are:

E(S1, S3) = 0.5 x E(R1, R2) x cos(resolver angle)

E(S2, S4) = 0.5 x E(R1, R2) x sin(resolver angle)

> ℹ️ **Note**
>
> The resolver must be tested together with a robot system to verify that the resolver also functions during battery mode.

Considerations

The following technical information must be considered before the installation:

- The maximum allowed resolver cable length is 30 m, from the resolver to the serial measurement board (SMB).

- A resolver cable consists of six wires. Two wires for excitation, and two wires each for the X and Y signals

- Use a shielded, AWG 24, max 55pF/m cable.

- To avoid disturbances in the signals due to magnetic fields generated by the brake it is recommended to use non-magnetic motor shaft.

> ℹ️ **Note**
>
> The unshielded part of the resolver cable must be as short as possible, less than 100 mm, and be well separated from the motor cables, more than 20 mm.

Resolver connection
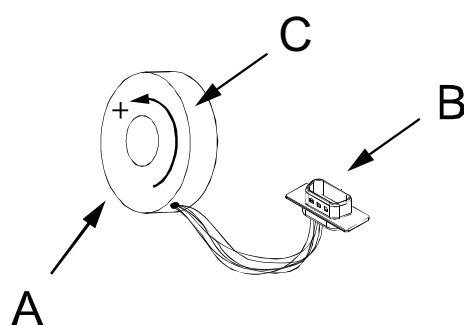
xx0400001172

A	Resolver
B	9 pin D-sub
C	Positive motor direction

Normally in ABB motors, resolvers are connected to the internal cable in robot by a 9 pin D-sub connector, with pins at the resolver side.

When the motor rotates in a positive direction, the resolver rotates mechanically in a negative direction, as the resolver is mounted at the opposite side of the drive shaft side.

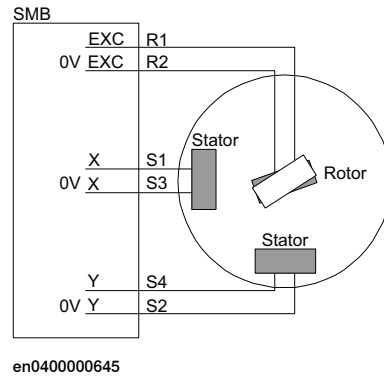To deliver electrically positive rotation the y-winding connection S2 and S4 has changed place.

9 pin D-sub	SMB input	Resolver connection	Color resolver wires
6	X	S1	Red
1	X 0V	S3	Black
7	Y	S4	Blue
2	Y 0V	S2	Yellow
3	EXC	R1	Red/White
8	EXC 0V	R2	Yellow/White

Continues on next page

en0400000645

Resolver direction

Motor angle	X (S1)	Y (S4)
0	Maximum in phase with EXC	0
+90	0	Maximum in phase with EXC

Commutation

Commutation can be done in several ways. The following method is one of the possible methods.

	Action	Info/Illustration
1	Turn the motor to commutation by feeding positive current into power winding S with T connected to ground (R is not connected). For detailed description, follow the first part of the procedure in *Manual commutation on page 56*.	The number of different commutation positions that the motor can turn to is the same as the number of pole pairs.
2	Select a resolver commutation position enabling the resolver cables to be routed in the best possible way.	
3	Feed a 4 kHz sinus signal to the EXC (R1) input of the resolver.	
4	Connect an oscilloscope to EXC (R1), X (S1) and Y (S4).	
5	Adjust the commutation position to +90 degrees +/-0.5 degrees by turning the resolver.	The Y (S4) signal should be at max output and with the same phase as the EXC (R1) feeding signal. The X (S1) signal should be 0.00 V xx0500001401

9.9 Serial measurement cables and connections

Overview

This section details the cables and connection between the resolver and the serial measurement board.

Signal classes

The cabling must comply with a valid signal class "measurement signals" see the cable requirements in the product manual for the robot controller. The enclosure for external serial measurement board/boards must comply with enclosure class IP54, in accordance with IEC 144 and IEC 529.

> **ℹ Note**
>
> It is very important that the noise level on the measurement signals from the additional axes is kept as low as possible, to prevent bad performance, that is, keep motor and resolver cables apart. Correct shielding and ground connections of cables, measurement boards and resolvers is essential.

Considerations

The **X**, **Y**, **0V X** and **0V Y**:

- Signals are used to connect resolvers to a serial measurement board.

The **EXC** and **0V EXC**:

- are used for common supply for all resolvers, parallel connected.

Resolver:

- 1 - 3, should always be connected to EXC 1.
- 4 - 7, should always be connected to EXC 2.

> **ℹ Note**
>
> Maximum allowed length on the serial measurement cable is 50 meters.
>
> For motor cables, see *Motors on page 108*.

9.9 Serial measurement cables and connections
Continued

Illustration DSQC 633C

xx2300000065

A	R2.SMB 1-2 (D-sub 15 socket)
B	R2.SMB 1-4 (D-sub 25 pin)
C	R2.SMB 3-6 (D-sub 25 socket)
D	R2.G
E	R2.SMB (D-sub 9 pin)

Connections to SMB DSQC 633C

Contact point	R2.G	R2.SMB	R2.SMB 1-2	R2.SMB 1-4	R2.SMB 3-6
1	+BAT	GND	GND	GND	GND
2	0V BAT	-	0V EXC2	X1	X4
3		0V	0V EXC1	Y1	Y4
4		SDO-N	Y7	X2	X5
5		SDI-N	X7	Y2	Y5
6		-	Y1	0V EXC1	0V EXC2
7		+24V	X1	0V EXC1	0V EXC2
8		SDO	-	0V EXC1	0V EXC2
9		SDI	EXC2	X3	X6
10			EXC1	Y3	Y6
11			0V Y7	X4	X3
12			0V X7	Y4	Y3
13			0V Y1	0V EXC2	0V EXC1
14			0V X1	0V X1	0V X4
15				0V Y1	0V Y4
16				0V X2	0V X5
17				0V Y2	0V Y5

Continues on next page

Contact point	R2.G	R2.SMB	R2.SMB 1-2	R2.SMB 1-4	R2.SMB 3-6
18				EXC1	EXC2
19				EXC1	EXC2
20				EXC1	EXC2
21				0V X3	0V X6
22				0V Y3	0V Y6
23				0V X4	0V X3
24				0V Y4	0V Y3
25				EXC2	EXC1

Explanation

Term	Description
SDO	serial communication output
SDI	serial communication input
+BAT	Battery +
0V BAT	Battery 0V
BATLD	Not to be used
BATSUP	Not to be used
EXC1	excitation power to resolver 1, 2, 3
EXC2	excitation power to resolver 4, 5, 6, (7)
+24V	24 V power
0 V	0 V power
X1	Input x-stator node 1
Y1	Input y-stator node 1

Illustration

The connection point on the resolver corresponds to the connection table above.

en0400000645

Example

To connect from resolver to SMB, use input 7 (node 7). Connect to contact R2.SMB 1-2.

Signals	Contact point SMB	Contact point resolver
EXC 2	9	3
EXC 2, 0 V	2	8
X7	5	6
X7, 0 V	12	1
Y7	4	7
Y7, 0 V	11	2

Index