

ROBOTICS

Application manual

Laser Tracker Calibration Interface



Trace back information:
Workspace Main version a515
Checked in 2023-06-05
Skribenta version 5.5.019

Application manual
Laser Tracker Calibration Interface

RobotWare 6.15.03

Document ID: 3HAC050957-001

Revision: F

The information in this manual is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any errors that may appear in this manual.

Except as may be expressly stated anywhere in this manual, nothing herein shall be construed as any kind of guarantee or warranty by ABB for losses, damage to persons or property, fitness for a specific purpose or the like.

In no event shall ABB be liable for incidental or consequential damages arising from use of this manual and products described herein.

This manual and parts thereof must not be reproduced or copied without ABB's written permission.

Keep for future reference.

Additional copies of this manual may be obtained from ABB.

Original instructions.

© Copyright 2008-2023 ABB. All rights reserved.
Specifications subject to change without notice.

Table of contents

Overview of this manual	7
1 Introduction to Laser Tracker Calibration	9
1.1 Principles of Laser Tracker Calibration	9
1.2 Prerequisites	10
1.3 Using RW Arc	12
1.4 Using Continuous Application Platform (CAP)	13
2 FlexPendant application	15
2.1 Overview	15
2.2 Calibration setup	16
2.2.1 Introduction to calibration setup	16
2.2.2 Defining a sensor	18
2.2.3 Defining a CalibID	19
2.2.4 Calibration setup wizard	20
2.3 Calibrating	22
2.3.1 Calibrating	22
2.3.2 Loading old calibration	25
2.4 Verifying	26
2.4.1 Verifying the calibration	26
2.5 Information views	28
2.5.1 Introduction to information views	28
2.5.2 Viewing the log	29
2.5.3 Viewing log details	30
3 RAPID interface	31
3.1 Using the RAPID modules without the FlexPendant interface	32
3.1.1 Preparations	33
3.1.2 Performing a calibration or verification	35
3.2 Variables	36
3.2.1 Variables in LTC_MySensordata (LTC_TemplateData)	36
3.3 System parameters	38
3.3.1 Topic Process	38
3.4 Instructions	42
3.4.1 LTCcalibrate	42
3.4.2 LTCverify	44
3.4.3 LTCGetEGMTool	46
4 Troubleshooting	47
4.1 Sensor types	47
4.1.1 My sensor type is not in the sensor type list when defining a sensor	47
4.2 Communication devices	48
4.2.1 Creating a new communication device	48
4.3 The precision of the calibration/verification is bad	49
4.3.1 The precision of the calibration/verification	49
4.3.2 Joint position error	51
4.3.3 General error from sensor	52
4.3.4 Backup	53
Index	55

This page is intentionally left blank

Overview of this manual

About this manual

This manual contains information about the *Laser Tracker Calibration* (LTC) interface built upon the RobotWare options *Sensor Interface* and *Optical Tracking* used with *ArcWare* or *Continuous Application Platform* (CAP) only. Unless something else is specified, *Laser Tracker Calibration* refers to the interface.

Usage

This manual is meant to be used as a description of how to use the *Laser Tracker Calibration* interface. It provides information about system parameters and RAPID components related to the use of laser trackers, as well as examples of how to calibrate them and verify the obtained precision.

Who should read this manual?

This manual is mainly intended for arc-welding robot operators having to deal with laser trackers.

Prerequisites

The reader should...

- be familiar with the IRC5 robot controller's FlexPendant
- be familiar with the RAPID programming language
- be familiar with system parameters and how to configure them.

References

References	Document ID
<i>Operating manual - IRC5 with FlexPendant</i>	3HAC050941-001
<i>Operating manual - RobotStudio</i>	3HAC032104-001
<i>Operating manual - Seam tracking with Weldguide III and MultiPass</i>	3HEA802921-001
<i>Technical reference manual - RAPID Instructions, Functions and Data types</i>	3HAC050917-001
<i>Technical reference manual - RAPID Overview</i>	3HAC050947-001
<i>Technical reference manual - System parameters</i>	3HAC050948-001
<i>Application manual - Controller software IRC5</i>	3HAC050798-001
<i>Application manual - MultiMove</i>	3HAC050961-001
<i>Application manual - Arc and Arc Sensor</i>	3HAC050988-001
<i>Application manual - Continuous Application Platform</i>	3HAC050990-001

Revisions

Revision	Description
-	Released with RobotWare 6.0.

Continues on next page

Revision	Description
A	Released with RobotWare 6.04. <ul style="list-style-type: none">• Added error handling for the RAPID instructions, see Instructions on page 42.
B	Released with RobotWare 6.06. <ul style="list-style-type: none">• Added new boolean in the LTC RAPID data module.
C	Released with RobotWare 6.07. <ul style="list-style-type: none">• Added section LTCGetEGMTool on page 46.
D	Released with RobotWare 6.10.02. <ul style="list-style-type: none">• Minor updates in Calibration setup wizard on page 20.
E	Released with RobotWare 6.14.01. <ul style="list-style-type: none">• Added limitation in section Eccentric sensor coordinate system on page 16.
F	Released with RobotWare 6.15.03. <ul style="list-style-type: none">• Updated some of the graphics.

1 Introduction to Laser Tracker Calibration

1.1 Principles of Laser Tracker Calibration

Principles

- **Maximum 3 calibrations per sensor:** A sensor may have more than one calibration (used in need of good accuracy in different places in space). The maximum allowed number is 3 calibrations per sensor.
- **Calibration plate:** A plate with a well-known geometry referenced by process parameters defined in PROC.cfg database and by a dedicated work object.
- **No simultaneous calibrations:** It is not possible to perform simultaneous sensor calibrations from the FlexPendant. Only one sensor per controller can be handled from the FlexPendant at a time. This includes calibration setup and verification.
- **MultiMove:** The program is designed to handle MultiMove systems. No restriction in the total amount of sensors that can be defined in a robot controller.
- **Execution from RAPID:** Calibrations and verifications can both be executed from the FlexPendant or using RAPID instructions. See [RAPID interface on page 31](#) for more information. The LTC__Manager.sys module allows several calibrations and/or verifications to be run simultaneously from RAPID in a MultiMove system comprising several sensors. But only one sensor can be handled at a time per motion task.

1 Introduction to Laser Tracker Calibration

1.2 Prerequisites

1.2 Prerequisites

Sensor prerequisites

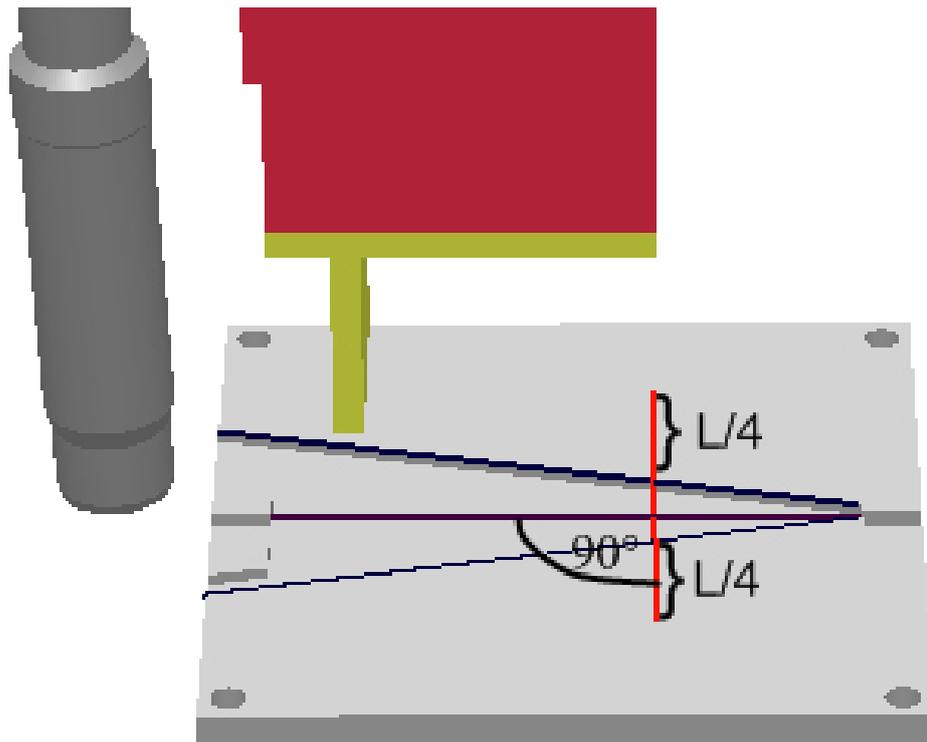
- The sensor is physically connected to the controller
- The sensor controller communication protocol is Itapp
- The sensor types are correctly defined in the process configuration database PROC.cfg



Note

Within the limitations indicated by the sensor manufacturer, the xxx angle between the sensor head and the torch is free. So is also the angle of the torch in the (xz) plane of the calibration plate. The y-axis of the welding gun should however be aligned with that of the plate. Other angles should be such that the sensor head is aligned with the welding gun both in the (xy) and (yz) planes of the sensor or the plate.

Illustration of angles



xx2300000872

Plate prerequisites

- At least one calibration plate of type ABB or SCOUT is mounted within the robots work area. The plate can be laid with any orientation including vertically or under a ceiling, but the robot work area around the plate shall be free of

Continues on next page

risk of collision. Possible errors may occur if the orientation of the plate does not allow reorientation of the robot tool during calibration and verification.

- The plate geometry is defined in the system parameter type *Calib_plate_type* in the topic *Process* (PROC.cfg).
- For each calibration plate geometry, one left, one right lap, and one butt joint must be defined in the sensor controller.
- The tracking point for these joints have to be located on the upper edge of the defined joints. The joint values should differ from those used during the welding process.



Note

The orientation of the tracker TCP's coordinate system always verifies:

- $y > 0$ refers to the left-lap joint
- $x > 0$ refers to the welding direction
- z is the result of the cross-product of the x and y vectors.

1 Introduction to Laser Tracker Calibration

1.3 Using RW Arc

1.3 Using RW Arc

Required options

The following options must be included in the RobotWare key for the robot system:

Description	Option number
Arc	633-1
Optical Tracking Arc	660-1

LTC__Tmp_CalPose and *LTC__Tmp_CalPos* must be defined as *Calib Variable Name (Pose and Pos)* in *Optical Sensor Properties* in the topic *Process* (system parameters), see *Application manual - Arc and Arc Sensor*.



Note

To add *Laser tracker Calibration* as a test view on the FlexPendant, open the **Control Panel, Additional text view**, and select the checkbox.

1.4 Using Continuous Application Platform (CAP)

Required options

The following options must be included in the RobotWare key for the robot system:

Description	Option number
Continuous Application Platform	624-1
Optical Tracking	813-1

LTC__Tmp_CalPose and *LTC__Tmp_CalPos* must be defined when running *CapLATRSetup* as *\CalibFrame* and *\CalibPos*, see *Application manual - Continuous Application Platform*.



Note

To add *Laser tracker Calibration* as a test view on the FlexPendant, open the **Control Panel, Additional text view**, and select the checkbox.

This page is intentionally left blank

2 FlexPendant application

2.1 Overview

General

The Laser Tracker Calibration application on the FlexPendant is used to manage laser tracking sensors.

- [Calibration setup on page 16.](#)
- [Calibrating on page 22.](#)
- [Verifying on page 26.](#)
- [Information views on page 28.](#)

2 FlexPendant application

2.2.1 Introduction to calibration setup

2.2 Calibration setup

2.2.1 Introduction to calibration setup

Overview

By creating your sensor system in LTC the system parameter configuration (topic *Process*) is updated. A RAPID data file for storing the calibration results is also created and loaded to the controller memory in the motion task specified for the used sensor.

Setting up a system

Follow these steps to set up a system:

- 1 Define a sensor, see [Defining a sensor on page 18](#).
- 2 Select a CalibID for that sensor, see [Defining a CalibID on page 19](#). Select a plate geometry and a work object for the plate.
- 3 Go through the calibration setup wizard, see [Calibration setup wizard on page 20](#).

In a more advanced system you might have more than one sensor defined and also more than one calibration for each sensor.

Eccentric sensor coordinate system

For some sensors from ServoRobot, the origin of the sensor coordinate system (SCS) is not centered in the sensor (blue arrow at X). Because of this displacement LTC cannot calibrate the sensor. To be able to calibrate the sensor using LTC the SCS has to be placed in the center of the sensor (green arrow, Y direction).

It is possible to move the origin of the SCS using the ServoRobot PC interface *WeldCom*. However, it is only possible to enter integer values for the displacement. That means that there can be up to 0.5 mm error for the position of the SCS.

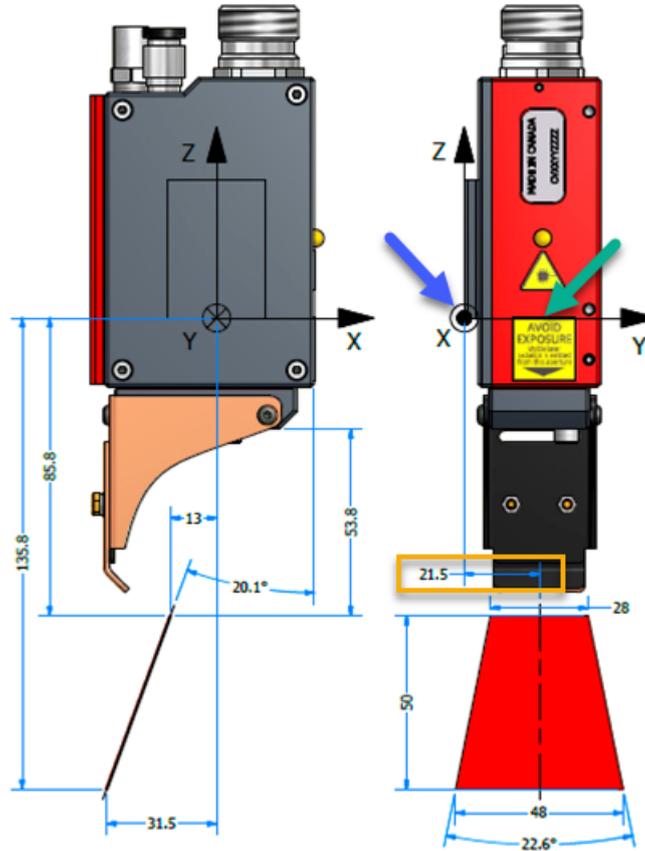
Continues on next page

2 FlexPendant application

2.2.1 Introduction to calibration setup

Continued

In the example below, the displacement of 21 or 22 has to be entered as displacement in WeldCom. For more information about WeldCom, contact ServoRobot.



xx2200001296

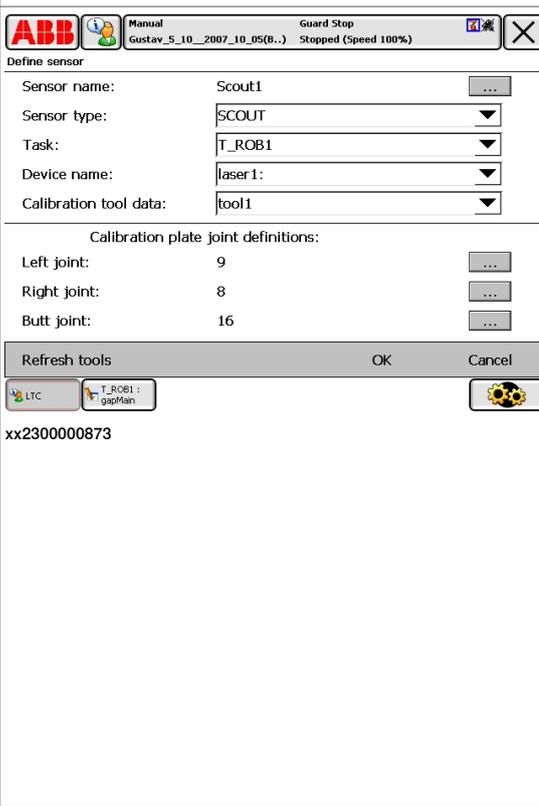
2 FlexPendant application

2.2.2 Defining a sensor

2.2.2 Defining a sensor

Defining a sensor

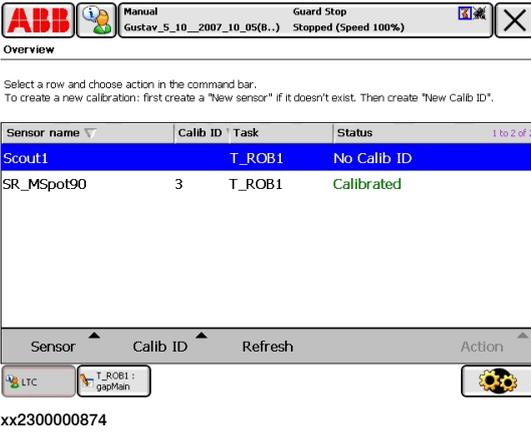
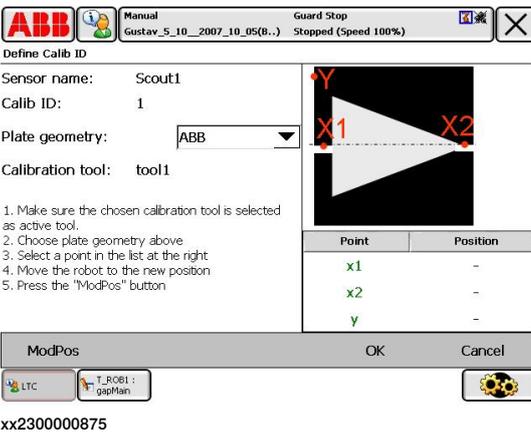
Use this procedure to define a sensor.

	Action	Note/illustration
1	On the FlexPendant ABB menu, tap Laser Tracker Calibration .	
2	On the menu Sensor , tap New .	
3	<p>Enter values in all fields.</p> <ol style="list-style-type: none"> 1 The sensor name is used by the program. 2 Sensor type defines the type of sensor. If your sensor type is not in the list, see My sensor type is not in the sensor type list when defining a sensor on page 47. 3 Select the motion task the sensor is used with. 4 Device name defines the communication device to be used with the sensor. To create a new communication device, see Communication devices on page 48. 5 Select the robot tool that the sensor should be used with. The sensor must be calibrated with this tool. 6 Set the joint definition number for each of the joint types to be used with the calibration plate. The number selected here should match the number defined within the sensor controller. 	
4	Tap OK when finished.	

2.2.3 Defining a CalibID

Defining a CalibID

Use this procedure to define a CalibID.

	Action	Note/illustration												
1	On the FlexPendant, use the Laser Tracker Calibration application.													
2	Select the sensor for which you would like to create a CalibID.	 <p>ABB Manual Gustav_5_10_2007_10_05(B...) Guard Stop Stopped (Speed 100%)</p> <p>Overview</p> <p>Select a row and choose action in the command bar. To create a new calibration: first create a "New sensor" if it doesn't exist. Then create "New Calib ID".</p> <table border="1"> <thead> <tr> <th>Sensor name</th> <th>Calib ID</th> <th>Task</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>Scout1</td> <td></td> <td>T_ROB1</td> <td>No Calib ID</td> </tr> <tr> <td>SR_MSpot90</td> <td>3</td> <td>T_ROB1</td> <td>Calibrated</td> </tr> </tbody> </table> <p>Sensor Calib ID Refresh Action</p> <p>LTC T_ROB1: gapMan</p> <p>xx2300000874</p>	Sensor name	Calib ID	Task	Status	Scout1		T_ROB1	No Calib ID	SR_MSpot90	3	T_ROB1	Calibrated
Sensor name	Calib ID	Task	Status											
Scout1		T_ROB1	No Calib ID											
SR_MSpot90	3	T_ROB1	Calibrated											
3	Tap the Calib ID menu and select a number. Up to 3 CalibID can be defined for each sensor.													
4	In the list Plate geometry , select the type of plate for this CalibID. An illustration of the selected plate is displayed to the right of the list.													
5	<p>Define the work object of the plate:</p> <ol style="list-style-type: none"> 1 Make sure that the correct tool is mounted on the robot. It must be the same tool as when defining the sensor, (calibration tool data). 2 In the Jogging window, activate the robot tool and Wobj0. Laser Tracker Calibration will show an error message if incorrect jogging parameters are selected before doing ModPos on the x1, x2, and y-positions on the calibration plate. 3 Select one of the positions in the list on the right. 4 Jog the robot to that position. 5 Tap ModPos to store the position. 6 Repeat for the other 2 positions. 	 <p>ABB Manual Gustav_5_10_2007_10_05(B...) Guard Stop Stopped (Speed 100%)</p> <p>Define Calib ID</p> <p>Sensor name: Scout1 Calib ID: 1 Plate geometry: ABB Calibration tool: tool1</p> <p>1. Make sure the chosen calibration tool is selected as active tool. 2. Choose plate geometry above 3. Select a point in the list at the right 4. Move the robot to the new position 5. Press the "ModPos" button</p> <table border="1"> <thead> <tr> <th>Point</th> <th>Position</th> </tr> </thead> <tbody> <tr> <td>x1</td> <td>-</td> </tr> <tr> <td>x2</td> <td>-</td> </tr> <tr> <td>y</td> <td>-</td> </tr> </tbody> </table> <p>ModPos OK Cancel</p> <p>LTC T_ROB1: gapMan</p> <p>xx2300000875</p> <p>Note</p> <p>Make sure that you are very accurate in your positioning of the robot. Any misplacement or displacement of the plate will affect the calibration result.</p>	Point	Position	x1	-	x2	-	y	-				
Point	Position													
x1	-													
x2	-													
y	-													
6	Tap OK when finished.	The program will calculate a work object for the plate. Any errors are displayed with a message box.												

2 FlexPendant application

2.2.4 Calibration setup wizard

2.2.4 Calibration setup wizard

Introduction to the calibration wizard

Calibration is set up with a wizard with seven steps. The wizard has three purposes:

- Step 1-3, defining robtargets for approach points and a starting position for calibration.
- Step 4-5, verifying that the calibration plate work object is correct.
- Step 6-7, verifying that the joint settings are good.

Defining approach points and a starting position for calibration

Three positions must be defined.

- Two approach positions, that is how the robot will approach the plate.
- One start position for calibration.

Selecting starting position

The following requirements apply for choosing the starting position, see figure below.

- The sensor should be positioned so that the laser beam partly falls over the calibration area of the plate.

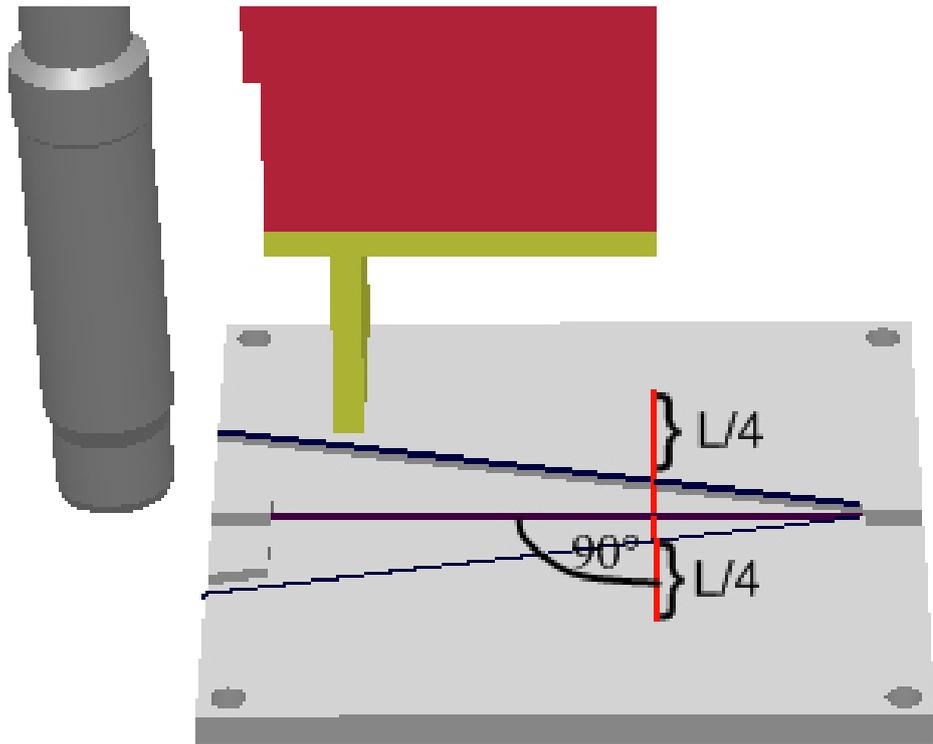


Note

About half of the field of view should fall outside the area!

- The y-axis of the sensor should be aligned with the y-axis of the plate, that is, the laser stripe is parallel to the y-axis of the calibration plate's work object.
- The sensor housing should be placed perpendicular to the calibration plate relative both the x-axis and the y-axis.
- To get the best results, the robot tool used for the calibration setup (for example welding gun) should have the same distance from and orientation relative the calibration plate as when running in production.

Continues on next page



xx2300000872

Verifying the origin of the plate work object

A RAPID module moving the robot to the origin of the work object is loaded to the controller in the specified motion task. Then the program pointer is set to that module.

Verify that the robot is at the real origin of the work object, within a vertical safety margin of a few millimeters (to avoid direct contact of the robot calibration tool with the plate).

Verify the joint settings

The last steps in the wizard will verify that the joint settings in the sensor controller are compatible with the plate and that the sensor works fine.

A RAPID module moving the robot looking for a joint of each type on the plate is loaded to the controller in the motion task specified for the used sensor and the program pointer is set to that module. Start its execution and supervise the action. The laser tracker beam is successively positioned above a right-lap joint, a left-lap joint and, if compatible with the calibration plate used, above a butt-joint. A reading of the respective joints is attempted before moving to the next joint.

The result of the action is summarized in the last step of the wizard.

2 FlexPendant application

2.3.1 Calibrating

2.3 Calibrating

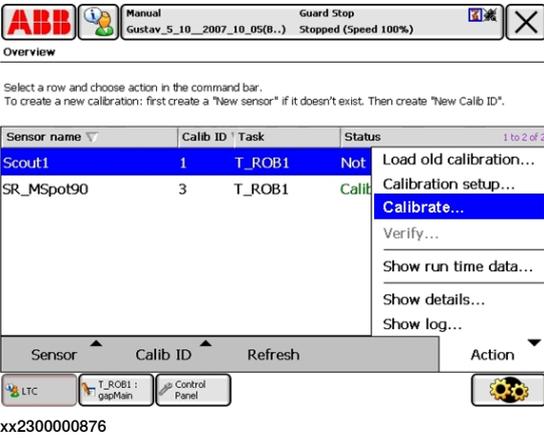
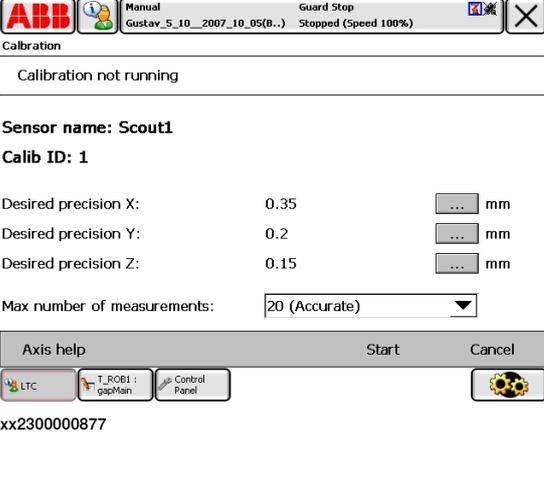
2.3.1 Calibrating

Prerequisites

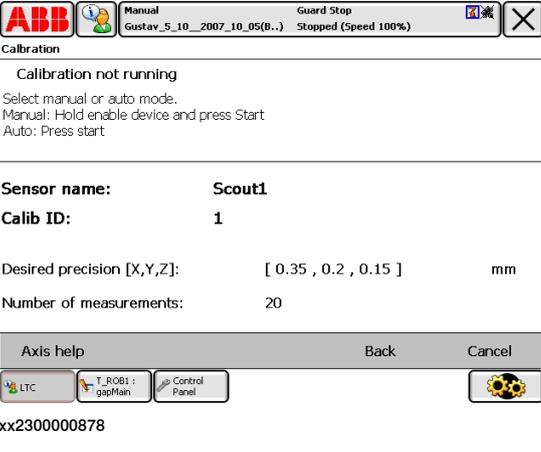
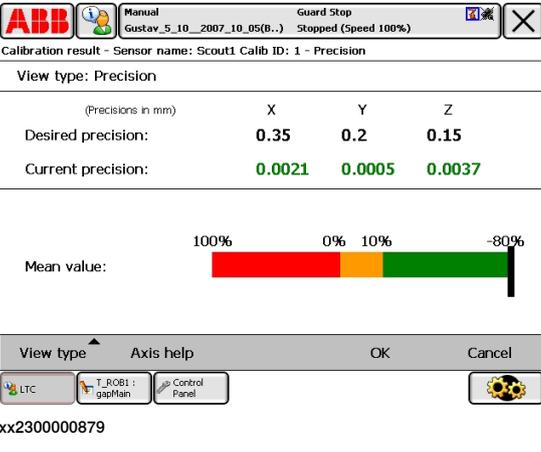
A system must be defined and the calibration setup wizard must be completed.

Calibrating

Use this procedure to calibrate.

	Action	Note/illustration												
1	On the FlexPendant, start the <i>Laser Tracker Calibration</i> application.													
2	Select the CalibID and sensor name that you want to calibrate and tap Calibrate on the Action menu.	 <p>The screenshot shows the 'Overview' screen of the calibration interface. At the top, there are status indicators for 'Manual' and 'Guard Stop'. Below that is a table with the following data:</p> <table border="1"> <thead> <tr> <th>Sensor name</th> <th>Calib ID</th> <th>Task</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>Scout1</td> <td>1</td> <td>T_ROB1</td> <td>Not</td> </tr> <tr> <td>SR_MSpot90</td> <td>3</td> <td>T_ROB1</td> <td>Calli</td> </tr> </tbody> </table> <p>A context menu is open over the 'Scout1' row, with 'Calibrate...' selected. Other options include 'Load old calibration...', 'Calibration setup...', 'Verify...', 'Show run time data...', 'Show details...', and 'Show log...'.</p>	Sensor name	Calib ID	Task	Status	Scout1	1	T_ROB1	Not	SR_MSpot90	3	T_ROB1	Calli
Sensor name	Calib ID	Task	Status											
Scout1	1	T_ROB1	Not											
SR_MSpot90	3	T_ROB1	Calli											
3	Select the precision and number of measurements. The desired precision depends on how the sensor will be used. Some applications need very high precision. The selected number will be used to decide if the calibration is accepted.	 <p>The screenshot shows the 'Calibration' screen. It displays the following settings:</p> <ul style="list-style-type: none"> Sensor name: Scout1 Calib ID: 1 Desired precision X: 0.35 mm Desired precision Y: 0.2 mm Desired precision Z: 0.15 mm Max number of measurements: 20 (Accurate) <p>Buttons for 'Start' and 'Cancel' are visible at the bottom.</p>												
4	Tap Start .													

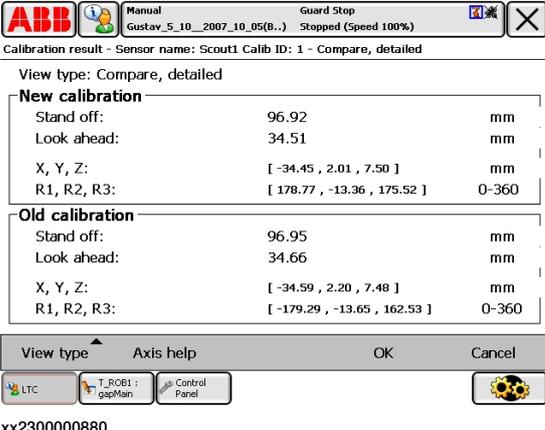
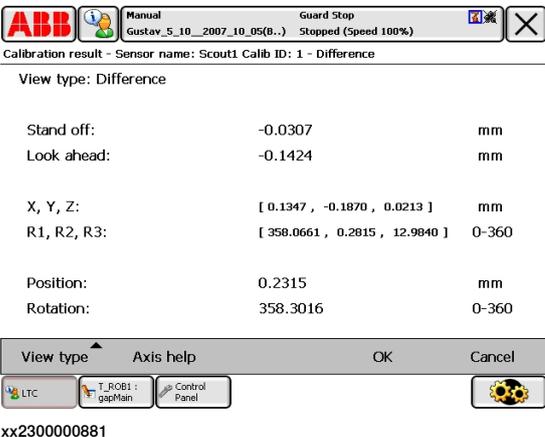
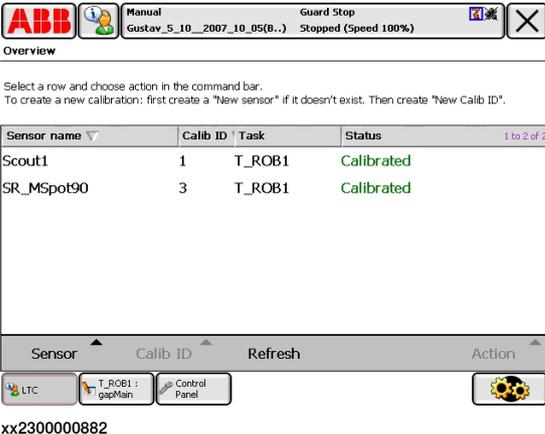
Continues on next page

	Action	Note/illustration												
5	<p>The necessary RAPID modules are loaded to the controller in the motion task specified for the used sensor and the program pointer is set. Press the Start button on the FlexPendant.</p>	 <p>Calibration</p> <p>Calibration not running</p> <p>Select manual or auto mode. Manual: Hold enable device and press Start Auto: Press start</p> <p>Sensor name: Scout1 Calib ID: 1</p> <p>Desired precision [X,Y,Z]: [0.35 , 0.2 , 0.15] mm Number of measurements: 20</p> <p>Axis help Back Cancel</p> <p>LTC T_ROB1 : gasMan Control Panel</p> <p>xx230000878</p>												
6	<p>Tap View type to display the result of the calibration.</p> <p>There are four views. The default view shows the precision, that defines if a calibration is accepted.</p>	 <p>Calibration result - Sensor name: Scout1 Calib ID: 1 - Precision</p> <p>View type: Precision</p> <p>(Precisions in mm)</p> <table border="1"> <thead> <tr> <th></th> <th>X</th> <th>Y</th> <th>Z</th> </tr> </thead> <tbody> <tr> <td>Desired precision:</td> <td>0.35</td> <td>0.2</td> <td>0.15</td> </tr> <tr> <td>Current precision:</td> <td>0.0021</td> <td>0.0005</td> <td>0.0037</td> </tr> </tbody> </table> <p>Mean value: 100% 0% 10% -80%</p> <p>View type Axis help OK Cancel</p> <p>LTC T_ROB1 : gasMan Control Panel</p> <p>xx230000879</p>		X	Y	Z	Desired precision:	0.35	0.2	0.15	Current precision:	0.0021	0.0005	0.0037
	X	Y	Z											
Desired precision:	0.35	0.2	0.15											
Current precision:	0.0021	0.0005	0.0037											

Continues on next page

2 FlexPendant application

2.3.1 Calibrating Continued

	Action	Note/illustration																																										
7	<p>Use the other views to check the differences or to compare with the new calibration result with the latest one saved.</p>	 <p>Calibration result - Sensor name: Scout1 Calib ID: 1 - Compare, detailed</p> <p>View type: Compare, detailed</p> <p>New calibration</p> <table border="1"> <tr><td>Stand off:</td><td>96.92</td><td>mm</td></tr> <tr><td>Look ahead:</td><td>34.51</td><td>mm</td></tr> <tr><td>X, Y, Z:</td><td>[-34.45 , 2.01 , 7.50]</td><td>mm</td></tr> <tr><td>R1, R2, R3:</td><td>[178.77 , -13.36 , 175.52]</td><td>0-360</td></tr> </table> <p>Old calibration</p> <table border="1"> <tr><td>Stand off:</td><td>96.95</td><td>mm</td></tr> <tr><td>Look ahead:</td><td>34.66</td><td>mm</td></tr> <tr><td>X, Y, Z:</td><td>[-34.59 , 2.20 , 7.48]</td><td>mm</td></tr> <tr><td>R1, R2, R3:</td><td>[-179.29 , -13.65 , 162.53]</td><td>0-360</td></tr> </table> <p>View type ^ Axis help OK Cancel</p> <p>LTC T_ROB1 : gapMain Control Panel</p> <p>xx2300000880</p>  <p>Calibration result - Sensor name: Scout1 Calib ID: 1 - Difference</p> <p>View type: Difference</p> <table border="1"> <tr><td>Stand off:</td><td>-0.0307</td><td>mm</td></tr> <tr><td>Look ahead:</td><td>-0.1424</td><td>mm</td></tr> <tr><td>X, Y, Z:</td><td>[0.1347 , -0.1870 , 0.0213]</td><td>mm</td></tr> <tr><td>R1, R2, R3:</td><td>[358.0661 , 0.2815 , 12.9840]</td><td>0-360</td></tr> <tr><td>Position:</td><td>0.2315</td><td>mm</td></tr> <tr><td>Rotation:</td><td>358.3016</td><td>0-360</td></tr> </table> <p>View type ^ Axis help OK Cancel</p> <p>LTC T_ROB1 : gapMain Control Panel</p> <p>xx2300000881</p>	Stand off:	96.92	mm	Look ahead:	34.51	mm	X, Y, Z:	[-34.45 , 2.01 , 7.50]	mm	R1, R2, R3:	[178.77 , -13.36 , 175.52]	0-360	Stand off:	96.95	mm	Look ahead:	34.66	mm	X, Y, Z:	[-34.59 , 2.20 , 7.48]	mm	R1, R2, R3:	[-179.29 , -13.65 , 162.53]	0-360	Stand off:	-0.0307	mm	Look ahead:	-0.1424	mm	X, Y, Z:	[0.1347 , -0.1870 , 0.0213]	mm	R1, R2, R3:	[358.0661 , 0.2815 , 12.9840]	0-360	Position:	0.2315	mm	Rotation:	358.3016	0-360
Stand off:	96.92	mm																																										
Look ahead:	34.51	mm																																										
X, Y, Z:	[-34.45 , 2.01 , 7.50]	mm																																										
R1, R2, R3:	[178.77 , -13.36 , 175.52]	0-360																																										
Stand off:	96.95	mm																																										
Look ahead:	34.66	mm																																										
X, Y, Z:	[-34.59 , 2.20 , 7.48]	mm																																										
R1, R2, R3:	[-179.29 , -13.65 , 162.53]	0-360																																										
Stand off:	-0.0307	mm																																										
Look ahead:	-0.1424	mm																																										
X, Y, Z:	[0.1347 , -0.1870 , 0.0213]	mm																																										
R1, R2, R3:	[358.0661 , 0.2815 , 12.9840]	0-360																																										
Position:	0.2315	mm																																										
Rotation:	358.3016	0-360																																										
8	<p>If the new calibration is approved, tap OK. The new calibration values will then be stored in RAPID variables.</p> <p> Note</p> <p>Before tapping OK, the values are only stored in temporary variables. When tapping OK the log will also be updated.</p>																																											
9	<p>The calibration is completed. The status of the corresponding CalibID and sensor name should be Calibrated. The sensor is now ready for use in production.</p>	 <p>Calibration result - Sensor name: Scout1 Calib ID: 1 - Difference</p> <p>Overview</p> <p>Select a row and choose action in the command bar. To create a new calibration: first create a "New sensor" if it doesn't exist. Then create "New Calib ID".</p> <table border="1"> <thead> <tr> <th>Sensor name</th> <th>Calib ID</th> <th>Task</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>Scout1</td> <td>1</td> <td>T_ROB1</td> <td>Calibrated</td> </tr> <tr> <td>SR_MSpot90</td> <td>3</td> <td>T_ROB1</td> <td>Calibrated</td> </tr> </tbody> </table> <p>Sensor ^ Calib ID ^ Refresh Action</p> <p>LTC T_ROB1 : gapMain Control Panel</p> <p>xx2300000882</p>	Sensor name	Calib ID	Task	Status	Scout1	1	T_ROB1	Calibrated	SR_MSpot90	3	T_ROB1	Calibrated																														
Sensor name	Calib ID	Task	Status																																									
Scout1	1	T_ROB1	Calibrated																																									
SR_MSpot90	3	T_ROB1	Calibrated																																									

2.3.2 Loading old calibration

Introduction

A previous calibration result that has been saved can be loaded to the robot controller for reuse.

Loading old calibration

Use this procedure to load a previous calibration.

	Action	Note/illustration
1	On the Action menu, tap Load old calibration .	
2	Select a calibration in the list and tap Show details to get more information.	Saved previous calibrations are listed chronologically.
3	Select a calibration in the list and tap Load to load the calibration results.	

2 FlexPendant application

2.4.1 Verifying the calibration

2.4 Verifying

2.4.1 Verifying the calibration

Introduction

The purpose of verification is to check if the calibration data is accepted. A RAPID module executes the verification of the calibration for the used sensor. No parameters need to be defined for the verification. The precision and number of measurements are read from the last performed calibration.

The verification uses three measurement sequences above both lap joints, respectively:

- laser beam aligned with Y axis of calibration plate
- laser beam rotated around its Z axis by an angle of +10 degrees with respect to the Y axis of calibration plate
- laser beam rotated around its Z axis by an angle of -10 degrees with respect to the Y axis of calibration plate

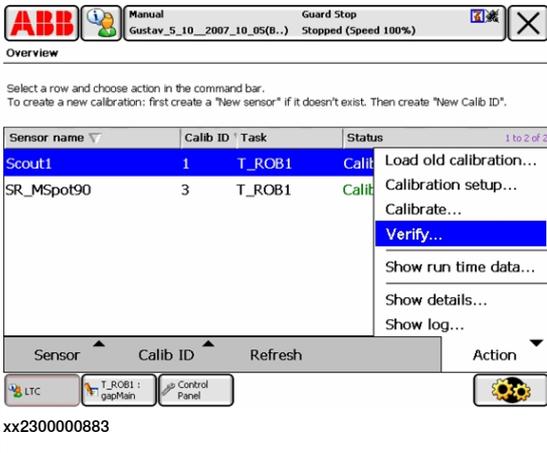
This sequence is repeated by moving the TCP of the sensor tool by 0.5 mm (SCOUT plate) or 1 mm (ABB plate) along the X axis of the plate as many times as the desired number of measurements input before calibration start.

Prerequisites

A calibration must have been done.

Verifying the calibration

Use this procedure to check the calibration data.

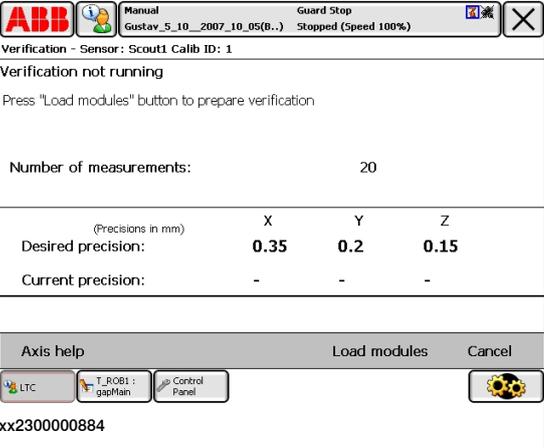
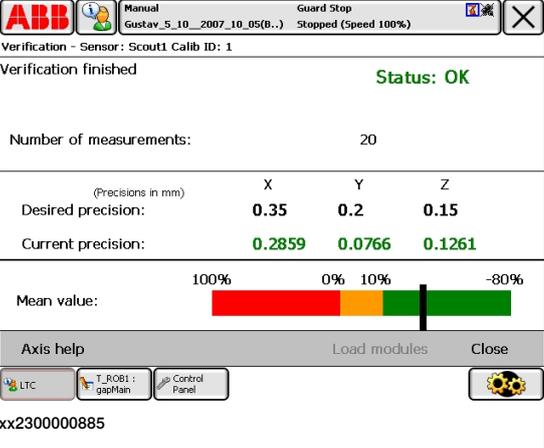
	Action	Note/illustration
1	On the FlexPendant, start the <i>Laser Tracker Calibration</i> application.	
2	Select the CalibID and sensor name to be verified.	
3	On the Action menu, tap Verify .	 The screenshot shows the ABB Laser Tracker Calibration Interface. At the top, there is a status bar with 'Manual' and 'Guard Stop' indicators. Below this is an 'Overview' section with instructions: 'Select a row and choose action in the command bar. To create a new calibration: first create a "New sensor" if it doesn't exist. Then create "New Calib ID".' A table lists two calibration entries: 'Scout1' with Calib ID '1' and 'SR_MSpot90' with Calib ID '3'. A context menu is open over the 'Scout1' row, with 'Verify...' selected. At the bottom, there are buttons for 'Sensor', 'Calib ID', 'Refresh', and 'Action'. The 'Action' button is highlighted, and the 'Verify...' option is visible in the dropdown menu.

Continues on next page

2 FlexPendant application

2.4.1 Verifying the calibration

Continued

	Action	Note/illustration												
4	Press the start button on the FlexPendant to run the verification.	 <p>Verification - Sensor: Scout1 Calib ID: 1</p> <p>Verification not running</p> <p>Press "Load modules" button to prepare verification</p> <p>Number of measurements: 20</p> <table border="1"> <thead> <tr> <th>(Precisions in mm)</th> <th>X</th> <th>Y</th> <th>Z</th> </tr> </thead> <tbody> <tr> <td>Desired precision:</td> <td>0.35</td> <td>0.2</td> <td>0.15</td> </tr> <tr> <td>Current precision:</td> <td>-</td> <td>-</td> <td>-</td> </tr> </tbody> </table> <p>Axis help Load modules Cancel</p> <p>LTC T_ROB1 : gasMan Control Panel</p> <p>xx230000884</p>	(Precisions in mm)	X	Y	Z	Desired precision:	0.35	0.2	0.15	Current precision:	-	-	-
(Precisions in mm)	X	Y	Z											
Desired precision:	0.35	0.2	0.15											
Current precision:	-	-	-											
5	<p>The results are displayed when the verification is finished. The log file is updated and the new precision values are written to the configuration (system parameters).</p> <p>If the precision is not acceptable, then do a new calibration. See Calibrating on page 22.</p>	 <p>Verification - Sensor: Scout1 Calib ID: 1</p> <p>Verification finished Status: OK</p> <p>Number of measurements: 20</p> <table border="1"> <thead> <tr> <th>(Precisions in mm)</th> <th>X</th> <th>Y</th> <th>Z</th> </tr> </thead> <tbody> <tr> <td>Desired precision:</td> <td>0.35</td> <td>0.2</td> <td>0.15</td> </tr> <tr> <td>Current precision:</td> <td>0.2859</td> <td>0.0766</td> <td>0.1261</td> </tr> </tbody> </table> <p>Mean value: 100% 0% 10% -80%</p> <p>Axis help Load modules Close</p> <p>LTC T_ROB1 : gasMan Control Panel</p> <p>xx230000885</p>	(Precisions in mm)	X	Y	Z	Desired precision:	0.35	0.2	0.15	Current precision:	0.2859	0.0766	0.1261
(Precisions in mm)	X	Y	Z											
Desired precision:	0.35	0.2	0.15											
Current precision:	0.2859	0.0766	0.1261											
6	Tap Close.													

2 FlexPendant application

2.5.1 Introduction to information views

2.5 Information views

2.5.1 Introduction to information views

Introduction

The logs are stored in the folder HOME:/LTC in the system. There is one log for each CalibID.

The log can be viewed on the FlexPendant or downloaded to a pc from the controller.

Log structure

The log header shows the name of the sensor, the CalibID used, and creation date. The log file is structured in as a table. Only the last 100 posts are saved in each log.

Runtime interface

The runtime interface gives real time control over the sensor in manual mode. Measurement values in y and z directions of the sensor tool are displayed continuously. The values are updated at the reading frequency of the sensor for a joint number.

It is possible to change the state of the sensor (on or off).

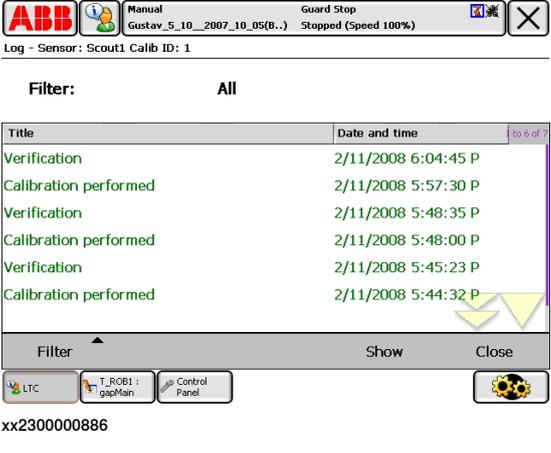
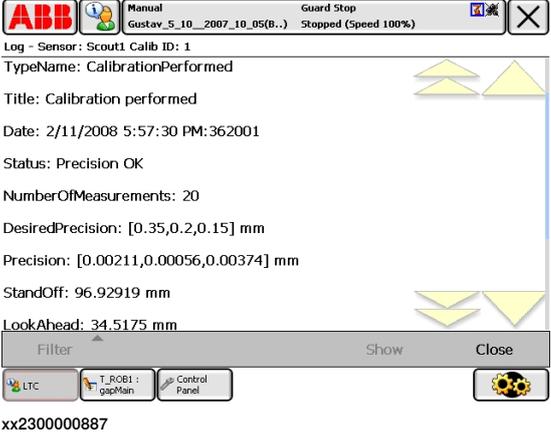
2.5.2 Viewing the log

Introduction to the log

The log view shows all events. Filters can be used to only show calibrations or verifications.

Viewing the log

Use this procedure to view the log.

	Action	Note/illustration														
1	On the FlexPendant, start the <i>Laser Tracker Calibration</i> application.															
2	Select the CalibID and sensor name.															
3	On the Action menu, tap Show log . The list of messages is displayed.	 <p>The screenshot shows the top status bar with 'ABB' logo, 'Manual' mode, 'Gustav_5_10_2007_10_05(8..)' text, and 'Guard Stop Stopped (Speed 100%)'. Below the status bar, it says 'Log - Sensor: Scout1 Calib ID: 1'. A 'Filter:' section is set to 'All'. A table lists events:</p> <table border="1"> <thead> <tr> <th>Title</th> <th>Date and time</th> </tr> </thead> <tbody> <tr> <td>Verification</td> <td>2/11/2008 6:04:45 P</td> </tr> <tr> <td>Calibration performed</td> <td>2/11/2008 5:57:30 P</td> </tr> <tr> <td>Verification</td> <td>2/11/2008 5:48:35 P</td> </tr> <tr> <td>Calibration performed</td> <td>2/11/2008 5:48:00 P</td> </tr> <tr> <td>Verification</td> <td>2/11/2008 5:45:23 P</td> </tr> <tr> <td>Calibration performed</td> <td>2/11/2008 5:44:32 P</td> </tr> </tbody> </table> <p>At the bottom of the log view, there are buttons for 'Filter', 'Show', and 'Close', along with 'LTC', 'T_ROB1: gapMan', and 'Control Panel' icons. The ID 'xx2300000886' is visible at the bottom.</p>	Title	Date and time	Verification	2/11/2008 6:04:45 P	Calibration performed	2/11/2008 5:57:30 P	Verification	2/11/2008 5:48:35 P	Calibration performed	2/11/2008 5:48:00 P	Verification	2/11/2008 5:45:23 P	Calibration performed	2/11/2008 5:44:32 P
Title	Date and time															
Verification	2/11/2008 6:04:45 P															
Calibration performed	2/11/2008 5:57:30 P															
Verification	2/11/2008 5:48:35 P															
Calibration performed	2/11/2008 5:48:00 P															
Verification	2/11/2008 5:45:23 P															
Calibration performed	2/11/2008 5:44:32 P															
4	Select a log and tap Show to see details.	 <p>The screenshot shows the same top status bar. Below it, it says 'Log - Sensor: Scout1 Calib ID: 1'. The event details are displayed:</p> <ul style="list-style-type: none"> TypeName: CalibrationPerformed Title: Calibration performed Date: 2/11/2008 5:57:30 PM:362001 Status: Precision OK NumberOfMeasurements: 20 DesiredPrecision: [0.35,0.2,0.15] mm Precision: [0.00211,0.00056,0.00374] mm StandOff: 96.92919 mm LookAhead: 34.5175 mm <p>At the bottom, there are buttons for 'Filter', 'Show', and 'Close', along with 'LTC', 'T_ROB1: gapMan', and 'Control Panel' icons. The ID 'xx2300000887' is visible at the bottom.</p>														

2 FlexPendant application

2.5.3 Viewing log details

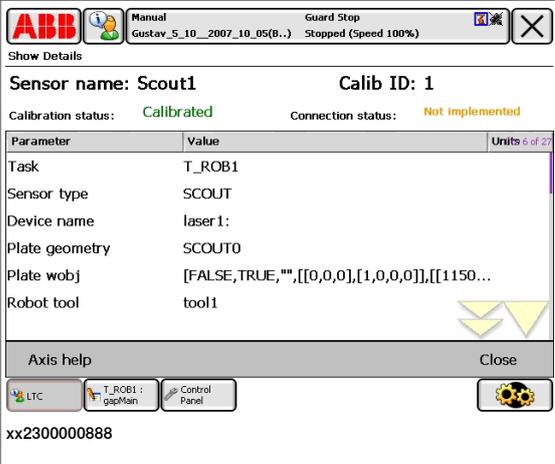
2.5.3 Viewing log details

Introduction

The details view shows all data and parameters for a CalibID.

Viewing log details

Use this procedure to select and view the details for a CalibID.

	Action	Note/illustration
1	On the FlexPendant, start the <i>Laser Tracker Calibration</i> application.	
2	Select the CalibID and sensor name.	
3	On the Action menu, tap Show details .	 <p>The screenshot displays the 'Show Details' window for a laser sensor. At the top, it shows the ABB logo and system status: 'Manual' (Gustav_5_10_2007_10_05(0..)) and 'Guard Stop' (Stopped (Speed 100%)). The sensor name is 'Scout1' and the Calib ID is '1'. The calibration status is 'Calibrated' and the connection status is 'Not implemented'. A table lists parameters: Task (T_ROB1), Sensor type (SCOUT), Device name (laser1), Plate geometry (SCOUT0), Plate wobj ([FALSE,TRUE,"",[[0,0,0],[1,0,0,0]],[[1150...]), and Robot tool (tool1). The interface includes 'Axis help' and 'Close' buttons, and a bottom bar with 'LTC', 'T_ROB1 : gapItem', 'Control Panel', and a gear icon. The ID 'xx2300000888' is visible at the bottom.</p>

3 RAPID interface

Introduction

Laser Tracker Calibration has a RAPID interface, that makes it possible to calibrate and verify the calibration on a regular basis during production. The module LTC__TemplateData.sys is used for generating specific data files when creating a new sensor.

Continues on next page

3 RAPID interface

3.1 Using the RAPID modules without the FlexPendant interface

3.1 Using the RAPID modules without the FlexPendant interface

Introduction

This section describes how to use the RAPID modules without the *Laser Tracker Calibration* application on the FlexPendant. This is useful when calibrating the sensor or verifying a calibration using RAPID.

The calibration setup is not described here. It should be done in advance from the FlexPendant application.

Since calibration and verification are similar, they are described together.

Continues on next page

3.1.1 Preparations

Prerequisites

Before running calibration or calibration verification using RAPID, the following must be done on the FlexPendant application:

- 1 Define the sensor. See [Defining a sensor on page 18](#).
- 2 Define a CalibID for that sensor. See [Defining a CalibID on page 19](#).
- 3 Set up a calibration. See [Calibrating on page 22](#).
- 4 Run a calibration. See [Calibrating on page 22](#).

System parameter, topic Process (PROC)

All parameters in SENSOR_TYPE, SENSOR_PHYSICAL, SENSOR_CALIB_ID, and CALIB_PLATE_TYPE must be defined correctly. This is done when the sensor is created, set up, and calibrated using the *Laser Tracker Calibration* application on the FlexPendant.

The system parameters can also be defined using FlexPendant Control Panel or RobotStudio Configuration Editor. The parameter settings can be exported as a file, PROC.cfg.

For more information about the parameters, see [Topic Process on page 38](#).

Sensor data file

A RAPID module named LTC__MySensorData, where *MySensor* is the sensor name, was created and loaded during the first setup of the sensor. That module must exist in the controller's memory before a calibration or a verification can be run without using the FlexPendant application. See [Variables on page 36](#) for information on the content of the sensor data file.

All the variable names include:

- *MySensor* is the sensor name.
- *X* is the CalibID number.

The following variables must be stored in LTC__MySensorData.sys for calibration or verification purposes:

Data type	Name	Required value
bool	LTC__MySensor_X_Keep_cal_Start	Yes, the default is FALSE
robtarg	LTC__MySensor_X_fast2plate	Yes
robtarg	LTC__MySensor_X_slow2plate	Yes
robtarg	LTC__MySensor_X_cal_Start	Yes
robtarg	LTC__MySensor_X_x1	No
robtarg	LTC__MySensor_X_x2	No
robtarg	LTC__MySensor_X_y	No
wobjdata	LTC__MySensor_X_Wobj	Yes
tooldata	LTC__MySensor_X_SenTool	No for calibration Yes for verification

Continues on next page

3 RAPID interface

3.1.1 Preparations

Continued

Data type	Name	Required value
pose	LTC__MySensor_X_CalPose	No
pos	LTC__MySensor_X_CalPos	No for calibration Yes for verification
pos	LTC__MySensor_X_Precision	No

3.1.2 Performing a calibration or verification

Introduction

Tracker calibration and verification can be initiated from customized RAPID modules by using the `LTC_Manager` system module.

Performing a calibration or verification

Use this procedure to calibrate and verify the sensor with the RAPID modules `LTC_Manager`.

	Action	Note/illustration
1	Make sure that the preparations are done, see Preparations on page 33 .	
2	Load the <code>LTC_Manager.sys</code> module to the motion task with the sensor to be calibrated or verified. During its execution, the module will automatically upload the system modules needed for calibration.	
3	Insert the instruction <code>CalibInstance</code> where the robot can reach the approach position <i>Fast2Plate</i> , to initiate a tracker calibration from a customized RAPID module. <pre>CalibInstance SensorName, \CalibID, \DesiredPrecision;</pre>	The execution of the instruction verifies that the obtained calibration precision is within the desired range. If not, RAPID execution will stop and an error message is displayed. A successful calibration is always followed by an automatic verification of the calibration. That is, as a result of at least one additional measurement it will be verified that the obtained average precision is within the desired range. See LTCcalibrate on page 42 .
4	To verify a tracker calibration (that is, no new calibration) from a customized RAPID module, insert the instruction <code>VerifInstance</code> where the robot can go to the approach position <i>Fast2Plate</i> . <pre>VerifInstance SensorName, \CalibID, \DesiredPrecision;</pre>	As mentioned above, at least one additional measurement it will be executed and it will be checked if the obtained precision is within the desired one. If not, RAPID execution will stop and an error message will be displayed. See LTCverify on page 44 .
5	The calibration/verification is now ready to be started.	

3 RAPID interface

3.2.1 Variables in LTC_MySensordata (LTC_TemplateData)

3.2 Variables

3.2.1 Variables in LTC_MySensordata (LTC_TemplateData)

Introduction

The table below describes all the variables defined in LTC__TemplateData.

MySensor refers to the name of the sensor to which a dedicated data file has been created.

1 means that CalibID = 1. In comparison, CalibID 2 suggests that the sensor name is followed by a 2 in each variable name instead.

Data type	Name	Used for
bool	LTC__My-Sensor_1_Keep_cal_Start	If the variable is set to TRUE, no automatic update of the calibration start position is performed. By default the value of this variable is FALSE, i.e. the start position is updated after aligning and centering the sensor relative to the sensor plate.
robtarget	LTC__MySensor_1_fast2plate	A remote position from where the robot can reach the calibration plate without collision. Defined in the Wobj LTC__ My-Sensor_1_Wobj
robtarget	LTC__ MySensor _1_slow2plate	A closer position from where the robot can reach the calibration plate without collision. Defined in the Wobj LTC__ MySensor _1_Wobj
robtarget	LTC__ MySensor _1_cal_Start	The start position of the calibration. <ul style="list-style-type: none">• The sensor should be positioned so that the laser beam partly falls over the calibration area of the plate. Note! About half of the field of view should fall outside the area.• The Y-axis of the sensor should be aligned with the Y-axis of the plate.• The welding gun or the chosen calibration tool (that is, the selected robot tool) should have the same distance and orientation to the plate as in normal production. Defined in the Wobj LTC__ MySensor _1_Wobj
robtarget	LTC__ MySensor _1_x1	A position on the X axis of the calibration plate, used for calculating the work object LTC__ MySensor _1_Wobj. This robtarget is defined in Wobj0.
robtarget	LTC__ MySensor _1_x2	A position on the X axis of the calibration plate, used for calculating the work object LTC__ MySensor _1_Wobj. This robtarget is defined in Wobj0.
robtarget	LTC__ MySensor _1_y	A position on the Y axis of the calibration plate, used for calculating the work object LTC__ MySensor _1_Wobj. This robtarget is defined in Wobj0.
wobjdata	LTC__ MySensor _1_Wobj	The Wobj of the calibration plate.

Continues on next page

Data type	Name	Used for
tooldata	LTC__ MySensor _1_SenTool	The tooldata of the sensor, used internally in LTC during verification.
pose	LTC__ MySensor _1_CalPose	The translation from the sensor tool TCP to the robot tool TCP.
pos	LTC__ MySensor _1_CalPos	The offset from sensor tool to point of normal production.
pos	LTC__ MySensor _1_Precision	The precision acquired during the latest calibration/verification.

3 RAPID interface

3.3.1 Topic Process

3.3 System parameters

3.3.1 Topic Process

Introduction

The following tables describe the system parameters relevant for *Laser Tracker Interface*. The types belong to the topic *Process*.

SENSOR_TYPE

Variable	Description
Name	The name of the sensor type (string).
Dimensions	The number of dimensions the sensor has, which is 2 or 3 (integer).
AngleCameraToLaser	The angle between the laser beam of the sensor and the camera axis (float).
Frequency	The reading frequency in Hz of the sensor, that is, the number of measurements the sensor captures every second (float).
WidthOfFieldClose	The width of the field-of-view at shortest distance (float).
WidthOfFieldFar	The width of the field-of-view at longest distance (float).
CloseStandOff	The stand-off distance from the z-ref of the sensor to its Close Field-of-View (num).
DepthOfField	The distance between shortest and longest range of view of the sensor (num).
OptTCPStandOff	The optimal TCP stand-off distance from the z-reference of the sensor to the optimal measuring point of the sensor (num).
FarStandOff	The stand-off distance from the z-ref of the sensor to its Far Field-of-View (num).
FrameAlignment	The alignment of the sensor measurement frame (string): <ul style="list-style-type: none">• laser: aligned with the laser beam• camera: aligned with the camera housing, not the laser beam (string)
FrameZOrientation	The orientation of the z component of the sensor measurement frame (num): <ul style="list-style-type: none">• +1 up, that is, "into" the camera• -1: down, that is, "out of" the camera
Brand	The brand of the sensor (string): <ul style="list-style-type: none">• ServoRobot• MetaScout

SENSOR_PHYSICAL

Variable	Description
Name	The name of the sensor for the purpose of calibration (string).
SensorType	The type of the sensor, see SENSOR_TYPE on page 38 , (string).
Task	The name of the motion task that uses this sensor (string).

Continues on next page

Variable	Description
DeviceName	The name of the device used for communication between the sensor and the robot controller (string).
RobTool	The name of the robot tool used with the sensor (string).
LeftLapJoint	The sensor controller's ID of the left lap joint for the calibration plate (integer).
RightLapJoint	The sensor controller's ID of the left right joint for the calibration plate (integer).
ButtJoint	The sensor controller's ID of the butt joint for the calibration plate (integer).
UsedForTracking	If the sensor is mounted as usual for tracking, that is, near the TCP of the robot tool that is used in a process application (for example arc welding), this parameter shall be set to <i>true</i> . There will be no difference compared to earlier RobotWare releases. If the sensor is not mounted for tracking, and points in an other direction than or is placed far from the robot tool TCP, then the parameter shall be set to <i>false</i> . The difference during calibration is, that no visual verification of the position of the calibration plate origo (x=0, y=0, z=0 in the calibration plate work object) will be performed. This step is skipped during the calibration setup to avoid collision with the sensor.

SENSOR_CALIB_ID

Variable	Description
Name	The name of the calibration instance, constituted by the sensor name for the purpose of calibration and the calibID (string).
SensorName	The name of the sensor, used for calibration. See SENSOR_PHYSICAL on page 38 , (string).
PlateGeometry	The name of the calibration plate used for the calibration instance (string).
LookAhead	The distance between the robot tool TCP and the sensor TCP at the CalPos position (float). Note! The sensor TCP is sometimes located within the unit, but the CapPos position corresponds to the location of the joint in normal production.
MaxSpeed	The maximum speed at which the sensor still can read the seam at a point and give a data feedback to the controller before the point is reached by the robot tool (float).
NumberOfMeasurements	The number of measurements carried out during the last calibration (integer).
DesiredPrecision	The precision required during the last calibration (string, in vector format: [0.532, 0.435, 0.345]).
LastCalibrated	The date and time of the last calibration (string).
LastVerified	The date and time of the last verification (string).
HasBeenSetup	True if the sensor has been setup, otherwise false (string, TRUE or FALSE).

Continues on next page

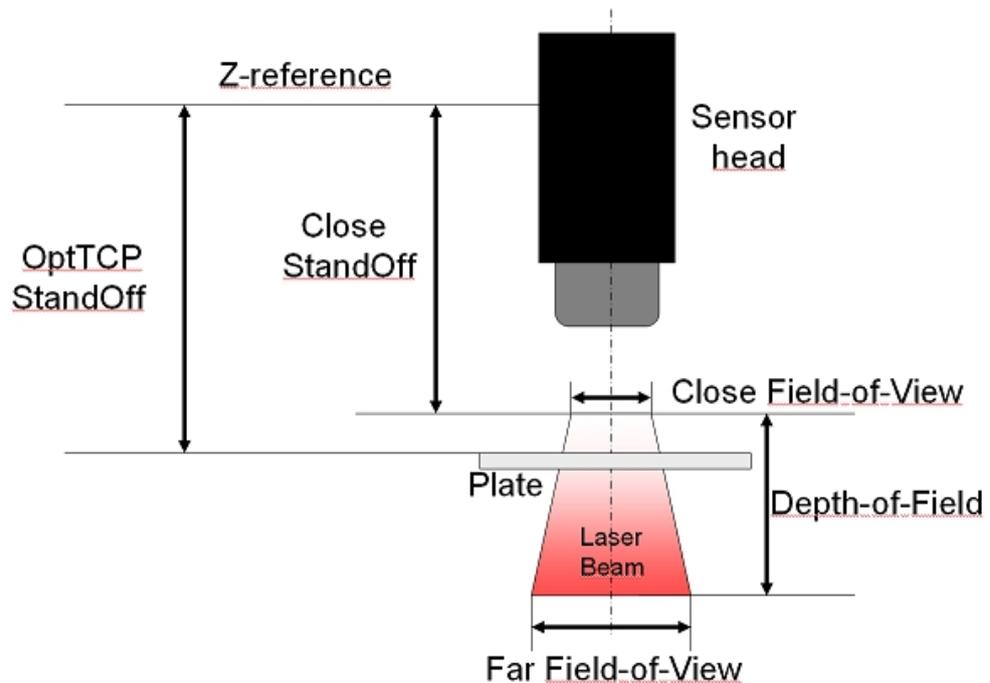
3 RAPID interface

3.3.1 Topic Process

Continued

CALIB_PLATE_TYPE

Variable	Description
Name	An abbreviation describing the name of the calibration plate available. For ease of use, the abbreviation can be terminated with a 0 or a 1 indicating the type of the calibration plate (string).
CalType	The type of the calibration plate (integer, 0 or 1).
Vertice_x	The x position of the calibration vertex in the coordinate system of the plate (float).
(Vertice_y)	The y position of the calibration vertex in the coordinate system of the plate (float)
CalAngle	The value of the calibration angle on the calibration plate (float)



xx230000889

Parameters described in `SENSOR_TYPE` (see [SENSOR_TYPE on page 38](#)) are commonly used for all sensor optical configurations in *Laser Tracker Calibration* and illustrated above. As sensor specifications vary between manufacturers, some parameter values may have to be deduced from others.

Field-of-view

By default, *Laser Tracker Calibration* will use the value entered as `FieldOfViewClose` in the type `SENSOR_TYPE`. If sensor specifications only refer to a nominal field-of-view value, store that value both as `FieldOfViewClose` and `FieldOfViewFar`, unless values can be measured.

Depth-of-Field

If sensor specifications refer to a useful depth-of-field at nominal stand-off in the form of an interval $[+z_{max}, -z_{min}]$ around the nominal stand-off position, calculate the depth-of-field as: $z_{max} - (-z_{min})$ and store that value as `DepthOfField`.

Continues on next page

Close Stand-off

If sensor specifications refer to a nominal stand-off distance, calculate the close stand-off distance as: $\text{nominal stand-off} - \text{Abs}(-z_{\text{min}})$ and store that value as *CloseStandOff*.

Example 1: Metavision sensor head MLP2/30

Data provided by the sensor manufacturer:

Field-of -view (horiz.) at nominal stand-off	30 mm
Nominal stand-off	75 mm
Useful depth-of-field at nominal stand-off	-30 to +45 mm

These data should appear in the SENSOR_TYPE field of PROC as:

<i>WidthOfFieldClose</i>	30
<i>WidthOfFieldFar</i>	30
<i>CloseStandOff</i>	45
<i>DepthOfField</i>	75
<i>OptTCPStandOff</i>	75

Example 2: ServoRobot sensor head SF-80/D

Data provided by the sensor manufacturer:

Depth-of-Field	85 mm
Width (close plane)	24 mm
Width (far plane)	36 mm
Close stand-off	97 mm

These data should appear in the SENSOR_TYPE field of PROC as:

<i>WidthOfFieldClose</i>	30
<i>WidthOfFieldFar</i>	30
<i>CloseStandOff</i>	45
<i>DepthOfField</i>	75
<i>OptTCPStandOff</i>	75

3 RAPID interface

3.4.1 LTCcalibrate

3.4 Instructions

3.4.1 LTCcalibrate

Usage

`LTCcalibrate` is used to enable creation of the instance of a tracker calibration from a customized RAPID module.

Basic example

```
LTCcalibrate "MSPOT90_PL ", \CalibID:=2\precision:=[0.5,0.4, 0.8];
```

The tracker of name `MSPOT90_PL` with calibration identity 2 is selected to run a calibration with precision [0.5, 0.4, and 0.8] respectively in the x, y, and z directions of the tracker's coordinate system.

Arguments

```
LTCcalibrate SenName, \CalibID, \precision;
```

`SenName`

Data type: `string`

The name of the sensor to be calibrated as declared in the system parameters (`PROC.cfg`).

`\CalibID`

Data type `num`, **default value:** 1

The calibration instance number as defined in `PROC.cfg`.

`\precision`

Data type `pos`, **default values** from `LTC`.

The desired precision respectively in the x, y, and z directions of the sensor tool expressed in millimeters as a vector. For example, [0.3, 0.2, 0.3]. When using [0,0,0] as desired precision, the values stored in `PROC.cfg` during the original calibration are reused automatically.

Program execution

`LTCcalibrate` is the name of a routine in the `LTC__Manager.sys` module. When executed, the routine first unloads data and calibration/verification system modules that may have existed in the controller's memory for the motion task used by the sensor, so that no error will happen when loading the system modules needed for a calibration in the motion task.

A data initialization is done where the `LTCcalibrate` arguments are used to read parameters from `PROC.cfg` and store their values in temporary variables defined in `LTC__TmpData.sys`. After a calibration, the obtained precision is compared to the desired one. If the obtained precision is good enough, the `LTC__Calibration.sys` module is unloaded and replaced with the `LTC__Verification.sys` module to run a verification of the newly performed calibration. Again, the desired precision is compared to the obtained precision.

Continues on next page

If the obtained precision differs from the desired, a recoverable error message is displayed on FlexPendant and RAPID execution is redirected to the error handler. Otherwise the uploaded system modules are unloaded from the motion task used by the sensor and RAPID execution resumes in the customized RAPID module.

Error handling

The following recoverable errors are generated and can be handled in an error handler. The system variable `ERRNO` will be set to:

Name	Cause of error
LTC_CALIB_ERR	This error is raised if the calibration of the sensor does not meet the imposed precision requirement.

Limitations

In a MultiMove system, the `LTC_Manager.sys` module has to be loaded in the respective motion tasks that own a laser tracker in order for `CalibInstance` commands to be executed from the respective motion tasks at the same time.

3 RAPID interface

3.4.2 LTCverify

3.4.2 LTCverify

Usage

LTCverify is used to enable the verification of a tracker calibration from a customized RAPID module.

Basic examples

```
LTCverify "MSPOT90_PL ", \CalibID:=2\precision:=[0.5,0.4, 0.6];
```

The tracker of name MSPOT90_PL with calibration identity 2 is selected to run calibration verification with the desired precision [0.5, 0.4 and 0.6] respectively in the x, y, and z directions of the tracker's coordinate system.

Arguments

```
LTCverify SenName, \CalibID, \precision;
```

SenName

Data type string

The name of the sensor to be calibrated as declared in the system parameters (PROC.cfg).

\CalibID

Data type num, default: 1

The calibration instance number as defined in PROC.cfg.

\precision

Data type pos, default: values from LTC

The desired precision respectively in the x, y, and z directions of the sensor tool expressed in millimeters as a vector. For example, [0.3, 0.2, 0.3]. When using [0,0,0] as desired precision, the values stored in PROC.cfg during the original calibration are reused automatically.

Program execution

LTCverify is the name of a routine in the LTC__Manager.sys module. When executed, the routine first unloads data and calibration/verification system modules that may have existed in the controller's memory for the motion task used by the sensor, so that no error will happen when loading the system modules needed for calibration verification in the motion task.

A data initialization is done where the LTCverify arguments are used to read parameters from PROC.cfg and store their values in temporary variables defined in LTC__TmpData.sys. After verification is done, the obtained precision is compared to the desired one.

If the obtained precision is good enough, the uploaded system modules are unloaded from the motion task used by the sensor and RAPID execution resumes in the customized RAPID module. If the obtained precision departs from the desired one, an error message is displayed on FlexPendant and RAPID execution is redirected to the error handler.

Continues on next page

Error handling

The following recoverable errors are generated and can be handled in an error handler. The system variable `ERRNO` will be set to:

Name	Cause of error
LTC_VERIFY_ERR	This error is raised if the verification of the sensor does not meet the imposed precision requirement.

Limitations

In a MultiMove system, the `LTC_Manager.sys` module has to be loaded in the respective motion tasks that own a laser tracker in order for `CalibInstance` commands to be executed from the respective motion tasks possibly at the same time.

3 RAPID interface

3.4.3 LTCGetEGMTool

RobotWare - OS

3.4.3 LTCGetEGMTool

Usage

`LTCGetEGMTool` converts the result of an LTC calibration, which is ready to use for CAP and RW Arc, into a tool, that can be used by EGM. The EGM tool frame is identical with the physical sensor measurement frame.

Basic examples

The following example illustrates the instruction `LTCGetEGMTool`.

Example 1

```
PERS tooldata tEGM;  
tEGM := LTCGetEGMTool("MSPOT90_PL ", 2);
```

The calibration data for the tracker `MSPOT90_PL` with calibration identity 2 is converted into a tool data that can be used with EGM path correction.

Arguments

```
LTCGetEGMTool(SenName, CalibID);
```

SenName

Data type: `string`

The name of the sensor to be calibrated as declared in the system parameters (`PROC.cfg`).

CalibID

Data type: `num`, **default value:** 1

The calibration instance number as defined in `PROC.cfg`.

Program execution

`LTCGetEGMTool` is the name of a routine in the `LTC_Manager.sys` module. When executed, the routine uses the LTC calibration result for the specified physical sensor and calibration identity (`tooldata SenName_CalibID_SenTool`, `pose SenName_CalibID_CalPose` and `pos SenName_CalibID_CalPos`) to calculate a sensor tool that is ready to use together with EGM Path correction.

Limitations

This instruction can only be used if a result from an LTC calibration is available.

4 Troubleshooting

4.1 Sensor types

4.1.1 My sensor type is not in the sensor type list when defining a sensor

Available sensor types

The list of available sensor types is created by reading the system parameter configuration for the type `SENSOR_TYPE` (topic *Process*). That is, the defined instances will be available in the sensor type list.

As a rule, most laser trackers for robotic arc welding applications available commercially from ServoRobot and Metavision-Scout work fine with the *Laser Tracker Calibration* interface. Off-lens sensors with circular beams produced by *Oxford Sensor Technology* are not supported by the interface.

Defining a sensor

To create new instances, modify the system parameters using RobotStudio or the FlexPendant, or by editing `PROC.cfg`. To use a new sensor type, make sure that the RAPID calibration modules can handle the sensor. Otherwise create new such modules.

4 Troubleshooting

4.2.1 Creating a new communication device

4.2 Communication devices

4.2.1 Creating a new communication device

Communication devices

Several communication devices can be necessary when using more than one sensor in a robot system. By default, `laser1:` is the only communication device predefined in the system input/output configuration file (`SIO.cfg`).

Creating a new communication device

To create a new communication device, modify the system parameters using RobotStudio or the FlexPendant, or by editing the `SIO.cfg` file. Add a new device name in the type *Transmission Protocol*, in the topic *Communication*.

Do a warm restart to load the modified system parameter configuration.



Note

Make sure the `SIO.cfg` file is updated with the correct baud rates for each COM port affected in the laser tracker systems.

4.3 The precision of the calibration/verification is bad

4.3.1 The precision of the calibration/verification

Poor calibration positions

Poor calibration precision can be caused by many reasons.

- The location of the calibration plate might have inadvertently been changed since the last calibration, or the plate might be damaged. Make sure that the plate is still at its original place and does not have any major scratches or other defects.
- The robot tool might be badly calibrated. Before using ModPos for the Wobj of the plate or calibrating the sensor, make sure that the robot tool itself is calibrated with good precision.
- The robot might be badly calibrated. Before using ModPos for the Wobj of the plate or calibrating the sensor, make sure that the robot is correctly calibrated. Update the robot revolution counters before continuing using your laser tracker system.
- The sensor might be loosely mounted. Make sure that the sensor is fastened well and not loose.



Note

As the robot stands still during calibration, in average the obtained precision during a calibration does not get much better by selecting a higher number of measurements. During verification of a calibration, the obtained precision can decrease slightly (that is, discrepancies increase a little) with the number of measurements performed but it is usually stable as worst measurements are rejected by mean deviation calculation and averaging is run over the remaining number of measurement points.

Inaccurate Wobj

One reason for poor calibration can be that the workobject (Wobj) for the plate is inaccurate. Make sure that the Wobj of the calibration plate is very accurately defined. Define the Wobj from the *Laser Tracker Calibration* graphical user interface in the view **Define CalibID**, or store the data to the correct RAPID variable in `LTC__MySensorData.sys`. The Wobj of a calibration is named `LTC__MySensor_X_Wobj`, where *MySensor* is the name of the sensor for the calibration instance, and X is the ID of the instantiation, referred to as *CalibID* (for example `LTC__Mspot_90_left_1_X_Wobj`).

Verifying the Wobj

Robot positions that define the plate Wobj can be verified quickly by using the **GOTO** function in the **Jogging** window.

The function moves the TCP straight from its current position to the selected robtarjet. Such movements can damage both a tool and the calibration plate if not with a safety distance between the TCP and the plate's surface.

Continues on next page

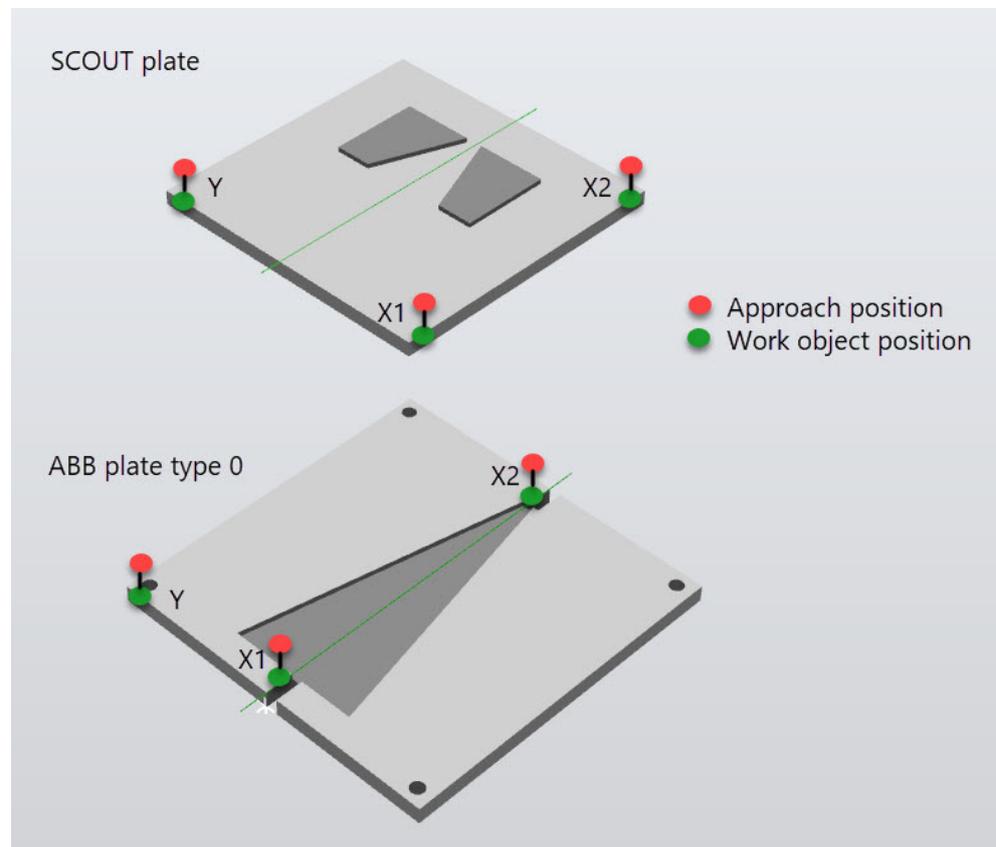
4 Troubleshooting

4.3.1 The precision of the calibration/verification

Continued

To use the **GOTO** function without any risk, first use a small RAPID program containing the following **MoveJ** instructions to verify that the approach positions are programmed correctly:

```
MoveJ approach_x1, v30, z10, tool1;  
MoveLx1,v30,fine, tool1;  
MoveL approach_x1, v30, z10, tool1;  
MoveJ approach_x2, v30,z10,tool1;  
MoveL x1, v30, fine, tool1;  
MoveL approach_x2,v30,z10, tool1;  
MoveJ approach_y, v30, z10, tool1;  
MoveLx1, v30, fine, tool1;  
MoveL approach_y, v30, z10, tool1;
```



xx2300000890

Using a 3-dimensional tracker

When using a 3-dimensional tracker with a short field of view, select a **Cal_start** position as close as possible to the calibration vertex of the plate so that all stripes of the tracker beam always cross both lap-joints of the plate at any time.



Note

In practice, the fifth stripe of a 3-dimensional tracker beam can sometimes lose contact with a lap-joint on the calibration plate, but measurement is still possible with four stripes only, without much effect on the obtained precision.

4.3.2 Joint position error

Joint position errors

The robot can have difficulties to reach programmed positions around the calibration plate due to their location relative to each other.

The *Laser Tracker Calibration* interface makes the robot movement as automatic as possible to minimize manual efforts from an operator.

Solving joint position errors

To solve joint position errors, consider the current orientation of the robot and see if another robot position could be used (ModPos) to calibrate the laser tracker.

Or, try changing the position of the calibration plate relative to the robot.

4 Troubleshooting

4.3.3 General error from sensor

4.3.3 General error from sensor

Solving sensor errors

Due to communication errors between sensor and controller, the background task *tLtcBgTsk* can stop. If so:

- Check that the baud rates are identical. Use the **Control Panel** on the FlexPendant or the **Configuration Editor** in RobotStudio. The type *Physical Channel* is in the topic *Communication*.
- Verify that the correct COM ports on the sensor's interface and on the controller are connected with each other.
- Check the joints definition in the sensor's PC interface.
- Restart the sensor's computer when all settings on the controller are correct.

4.3.4 Backup

Creating a backup

Making a backup of your robot system from FlexPendant is a cautious step to save your calibration result together with other process and sensor data. In case of problems affecting the general settings of the robot system, the saved calibration can be retrieved anytime by loading the backup to the robot controller's memory.

This page is intentionally left blank

Index

A

angles, 10

C

CALIB_PLATE_TYPE, 40

CalibID

defining, 19

calibrating, 22

calibration

loading old, 25

running, 22

setting up, 20

verifying, 26

CAP

required options, 13

I

instructions

LTCcalibrate, 42

LTCGetEGMTool, 46

LTCverify, 44

L

LTCcalibrate, instruction, 42

LTCGetEGMTool, 46

LTCverify, instruction, 44

O

options

CAP, 13

RW Arc, 12

P

plate prerequisites, 10

prerequisites, 10

PROC.cfg, 38

Process

topic, 38

R

RW Arc

required options, 12

S

sensor

defining, 18

SENSOR_CALIB_ID, 39

SENSOR_PHYSICAL, 38

SENSOR_TYPE, 38

system parameters

Process, 38

V

variables, 36

verifying, 26



ABB AB

Robotics & Discrete Automation

S-721 68 VÄSTERÅS, Sweden

Telephone +46 10-732 50 00

ABB AS

Robotics & Discrete Automation

Nordlysvegen 7, N-4340 BRYNE, Norway

Box 265, N-4349 BRYNE, Norway

Telephone: +47 22 87 2000

ABB Engineering (Shanghai) Ltd.

Robotics & Discrete Automation

No. 4528 Kangxin Highway

PuDong New District

SHANGHAI 201319, China

Telephone: +86 21 6105 6666

ABB Inc.

Robotics & Discrete Automation

1250 Brown Road

Auburn Hills, MI 48326

USA

Telephone: +1 248 391 9000

abb.com/robotics