HxGN Robotic Automation

Expert User Manual

Expert User Manual
12 October 2023
Created with Version 2023.1

Information about this document

Parts of this document may be automatically translated.

Document History

Version	Date	Author(s)	Modifications / Remarks
1.0	26.07.2023	Beny Mokhov, Alejandro Muñoz, Itzik Neman, Corrado Pavan, Bart Vermeire	First emission
1.1	27.07.2023	Alejandro Muñoz	Corrected PC-DMIS column in table 10.4
1.2	07.08.2023	Corrado Pavan	Added missing information about workstation (chapter 4.4) and its setup for robot alignment process (chapter 7.4).
1.3		Corrado Pavan	• Corrected error on chapter 8.8.1 • Removed chapter referring to older versions upgrade • Added warning on chapter 10.3.1

CONTENTS

1 Introduction

HxGN Robotic Automation is a software application, delivered as part of a Robot Based Inspection Cell, which is an integrated system composed of multiple HW and SW components connected and coordinated to perform inspection tasks.

HxGN Robotic Automation allows the user to create an integrated solution for part inspection, combined with a Metrology SW Package (Inspire, PC-DMIS or PolyWorks) and connected to Automation Hardware Devices (Robots, PLCs, Sensors, Trackers, Part Holders).

The image below shows the different components that are part of a Common Robot Based Inspection cell, all interconnected to create a Robotic Solution for Part Inspection.

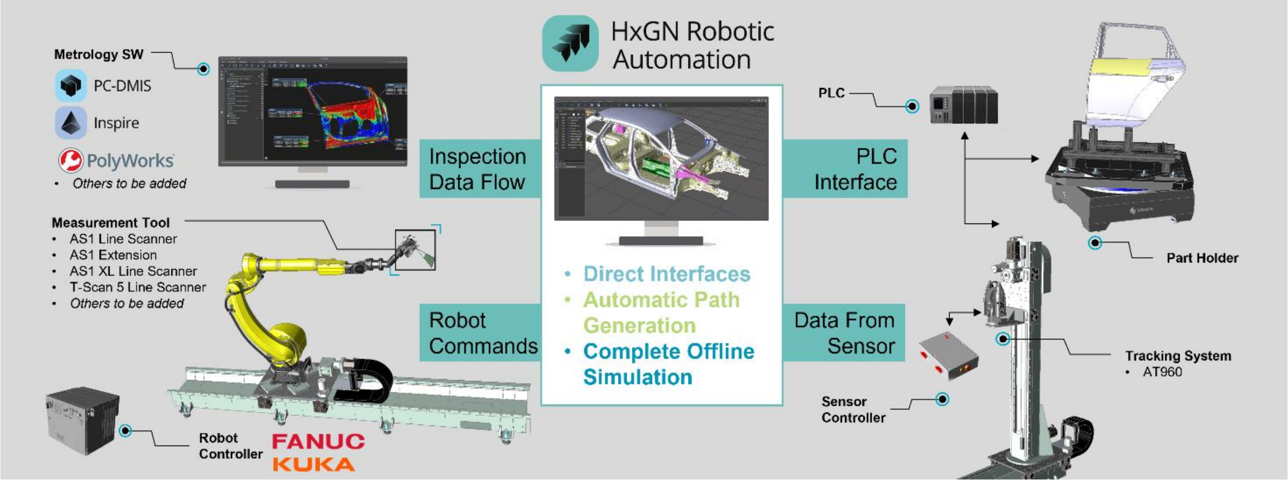


Figure 1-1*: HxGN Robotic Automation connection scheme*

HxGN Robotic Automation supplies the toolset to Program and Control Robotic Cells designed for Metrology and Part Inspection. It Simplifies the interface with the Cell components and the Robot Programming perfecting the Data Acquisition Process for Smart Manufacturing.

Besides its Ease of Use and Interoperability, which makes it a Universal Robot Programming Platform, one of its key differentiators is its planning capabilities, with a Path Planning Engine that is tailored to support the main HW Measurements Devices for part inspection and their Measurement Strategies needed for best data acquisition.

HxGN Robotic Automation Key Features are:

- Direct Hardware interface with Robot, Sensors, and other devices, explained in a dedicated Service Installation Manual for use by system integrators.
- Machine Modelling, explained in chapter 5.1
- Cell Modelling and Layout, explained in chapter 5.2
- Robot Calibration and Cell alignment capabilities, explained in chapter 7
- Inspection Programming with Automatic Robot Path Generation based on Inspection Plans or Part Volume /Region of Interest with embedded Real-Time Collision Detection and Avoidance, explained in chapter 8
- Robot program Execution, explained in chapter 9
- Integration with Metrology Software with Realistic Data Simulation, explained in chapter 0

The image below shows the complete user workflow to execute a robotic inspection task, which we can break into 4 steps:

Prepare:

Preparation of the Measurement task Based on Inspection Plan in Metrology Software

Plan:

Automatic generation of the Robot Program for Part Inspection in HxGN Robotic Automation

Execute:

Online Program Execution for Robot Control and Measurement Data Acquisition

Measure:

Measurement Analysis in Metrology SW based on data received from HxGN Robotic Automation

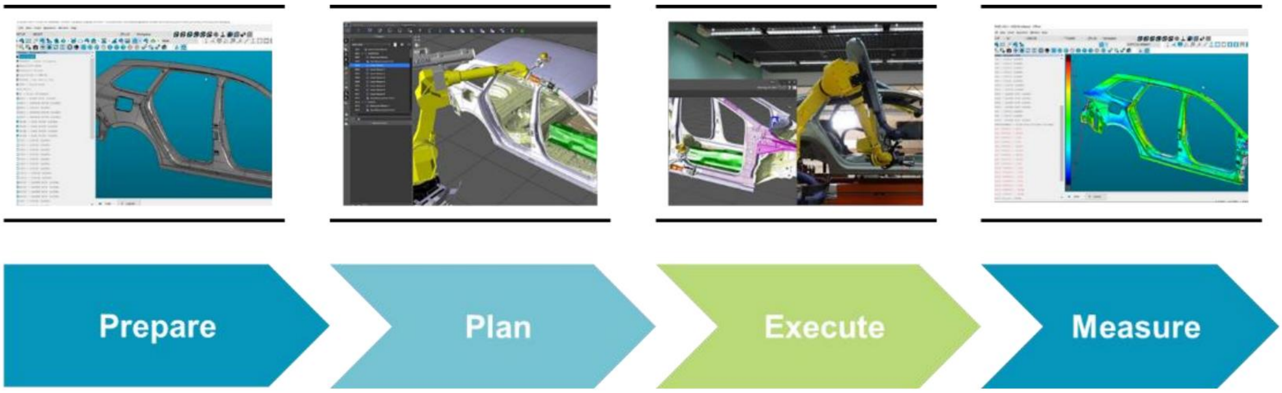


Figure 1-2: *HxGN Robotic Automation User Workflow*

HxGN Robotic Automation closes the gap related to Automated Data Inspection Data Acquisition for Quality Assurance and Process Optimization.

As part of the Digital Transformation, all this Data (Metrology Data) is used to populate and enrich a Quality Data Management System which makes the best use of the Quality and Production Data to improve the PLM (Product Lifecycle Management) process from start to end.

2 Recommended System Requirements

The following table shows the recommended requirements to run HxGN Robotic Automation on a computer.

Operating system	HxGN Robotic Automation operates under 64-bit Windows 11 and Windows 10. No other operating systems are supported.
RAM	64 GB of RAM or higher
CPU	2 GHz or higher quad core processor **Recommended** Intel Core i9 9th Generation or higher (for example, i9-9xxxHx) 8 Cores hyper-threading enabled
Graphics	NVIDIA RTX series graphics card equipped with 20 GB of memory (NVIDIA certified cards and drivers)
Hard Drive	SSD drive at least 2 TB, 5 GB of free hard drive space for HxGN Robotic Automation applications
Display	Screen resolution of 1920 x 1080 or higher
Connectivity	One Gigabit Ethernet port. Three Ethernet ports may be required for specific online installations.
Browsers	Microsoft Edge or Chrome
Anti-virus software	Hexagon used the Sophos anti-virus tool to test HxGN Robotic Automation 2023.1. The user will need to confirm the performance of any other anti-virus tool. http://sophos.com/products/enterprise/endpoint/security-and-control/

3 Installation and licensing

Configuration files used in HxGN Robotic Automation are saved under:

“C:\ProgramData\Hexagon\HxGN.RA.GenericHost”
“C:\ProgramData\Hexagon\RA”
“C:\ProgramData\Hexagon\TrackerBridge”

After upgrading HxGN Robotic Automation version, it is recommended to compare the configuration files above with those saved under:

“C:\Program Files\RA”

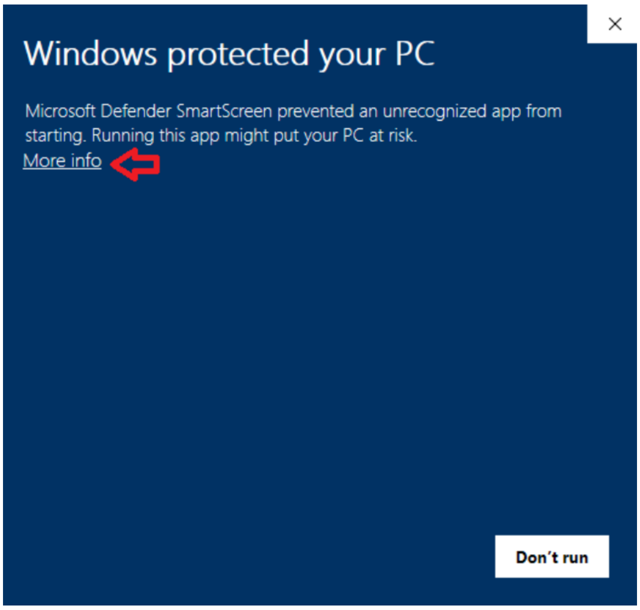
to verify any changes and ensure the proper functioning of HxGN Robotic Automation.

In case there are differences the latter files must be used but keeping the correct IP addresses.

3.1 Installation workflow

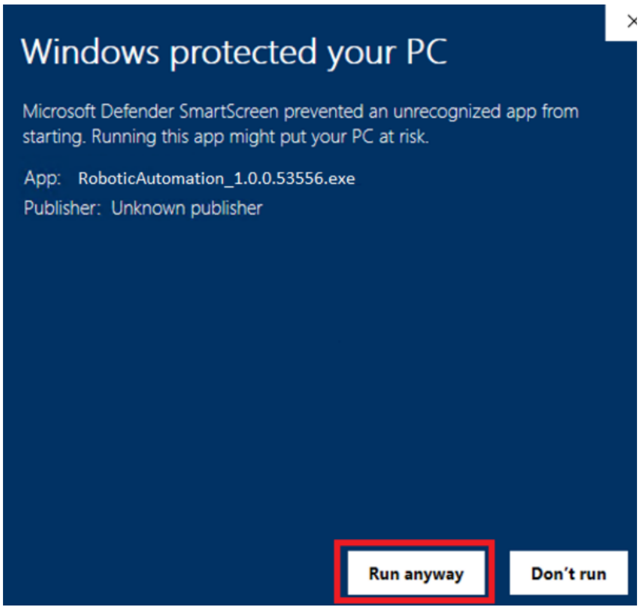
To use HxGN Robotic Automation for offline programming it is needed to install the Robotic Automation main application by following these steps.

Right click on HxGN Robotic Automation installer and click run as administrator. In case you get a warning message from Window as follows, please click on **More Info**, and then click **Run anyway** button.



Windows protected your PC

Microsoft Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

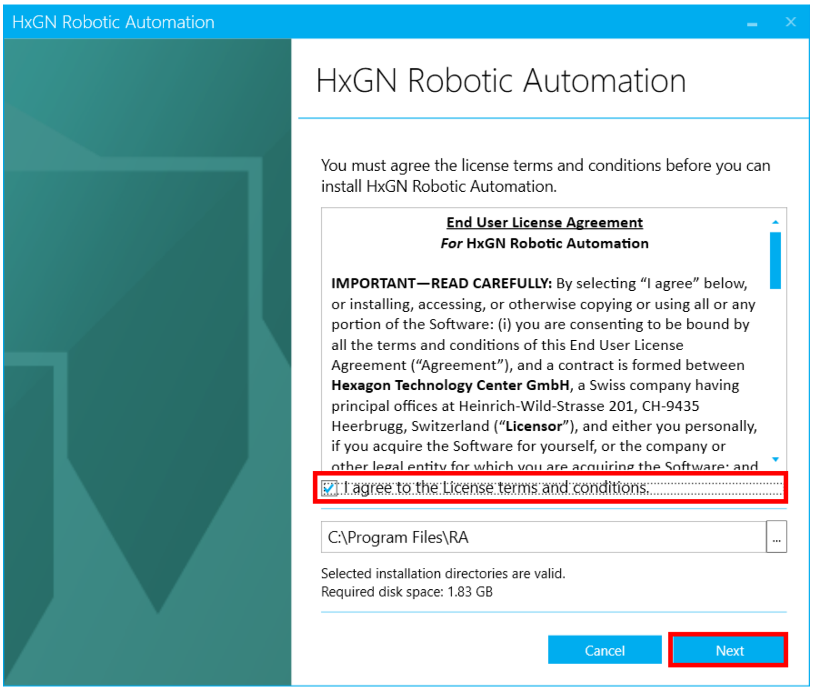


Windows protected your PC

Microsoft Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

App: RoboticAutomation_1.0.0.53556.exe
Publisher: Unknown publisher

When the installer dialog box pops up, check the checkbox to agree on the license terms and condition and then click on **Next**.



HxGN Robotic Automation

You must agree the license terms and conditions before you can install HxGN Robotic Automation.

End User License Agreement
***For* HxGN Robotic Automation**

IMPORTANT—READ CAREFULLY: By selecting "I agree" below, or installing, accessing, or otherwise copying or using all or any portion of the Software: (i) you are consenting to be bound by all the terms and conditions of this End User License Agreement ("Agreement"), and a contract is formed between **Hexagon Technology Center GmbH**, a Swiss company having principal offices at Heinrich-Wild-Strasse 201, CH-9435 Heerbrugg, Switzerland ("**Licensor**"), and either you personally, if you acquire the Software for yourself, or the company or other legal entity for which you are acquiring the Software; and

☑ I agree to the License terms and conditions.

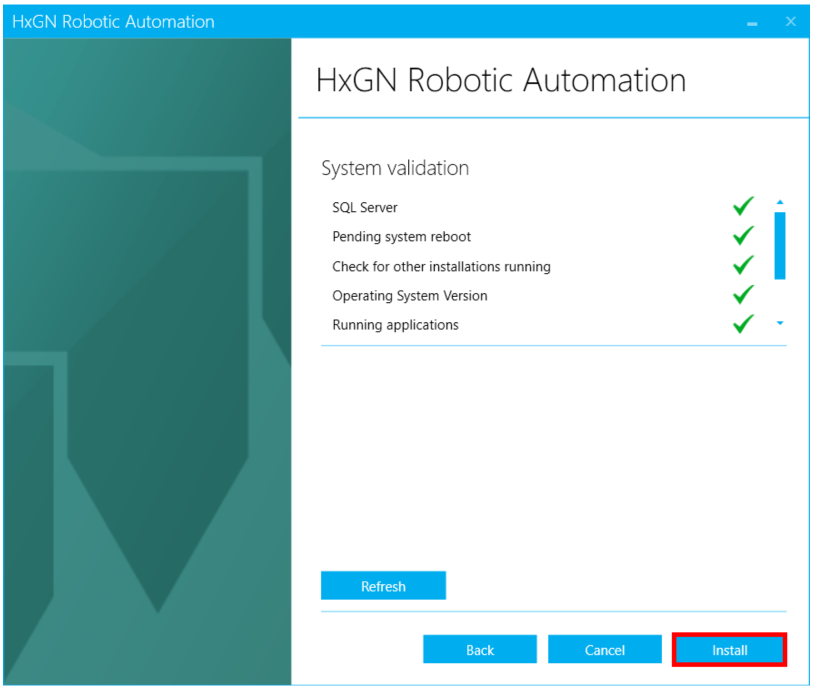
C:\Program Files\RA

Selected installation directories are valid.
Required disk space: 1.83 GB

Robotic Automation install package will perform a System validation by checking:

- SQL server (green checkmark if installed). If not installed, it will be automatically installed.
- Pending system reboot (green checkmark if there is no system reboot pending).
- Check for other installation running (green checkmark if there is no other installation running on your system).
- Operating System version (green checkmark if Microsoft Windows OS is supported).
- Microsoft Excel is recommended but not mandatory.

Then click **Install**.

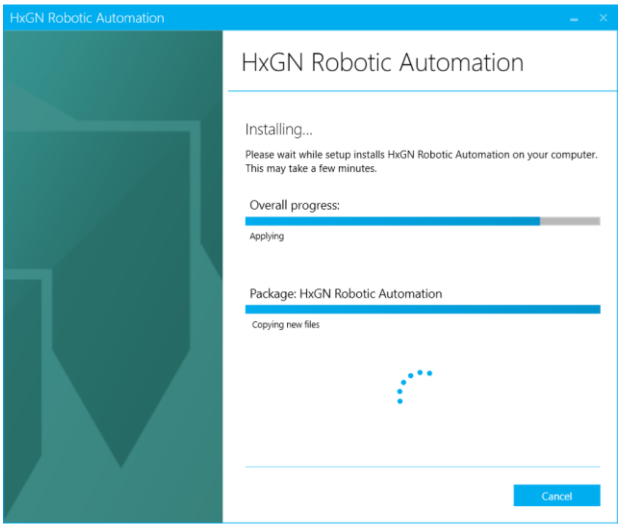


HxGN Robotic Automation

System validation

SQL Server ✓
Pending system reboot ✓
Check for other installations running ✓
Operating System Version ✓
Running applications ✓

After the installation is completed the dialog box can be closed and HxGN Robotic Automation shortcut icon will be created on your Desktop.



HxGN Robotic Automation

HxGN Robotic Automation

Installing...

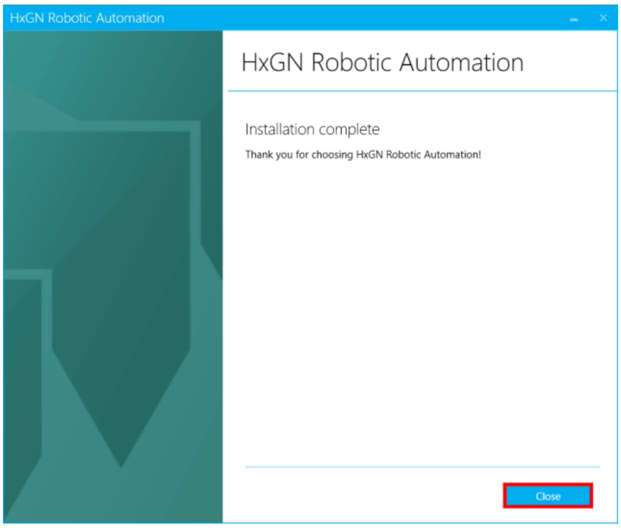
Please wait while setup installs HxGN Robotic Automation on your computer. This may take a few minutes.

Overall progress:

Applying

Package: HxGN Robotic Automation

Copying new files



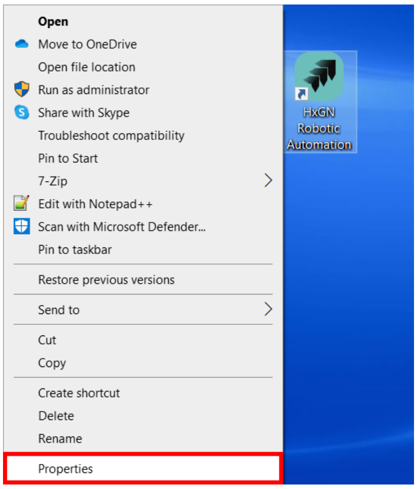
HxGN Robotic Automation

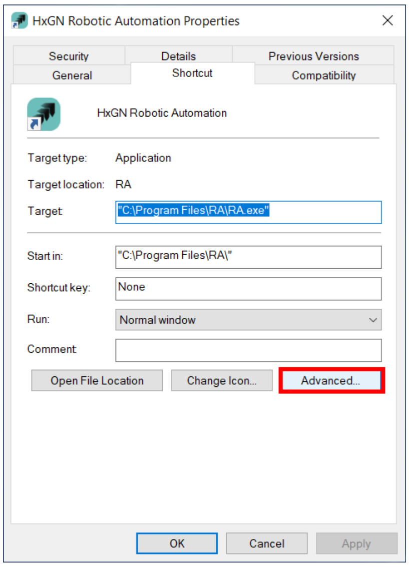
HxGN Robotic Automation

Installation complete

Thank you for choosing HxGN Robotic Automation!

It is always recommended to execute HxGN Robotic Automation as administrator. Right-click on the desktop shortcut and select **Properties**. Check the checkbox in the shortcut advanced properties as shown in the images below.





HxGN Robotic Automation Properties

HxGN Robotic Automation

Target type: Application

Target location: RA

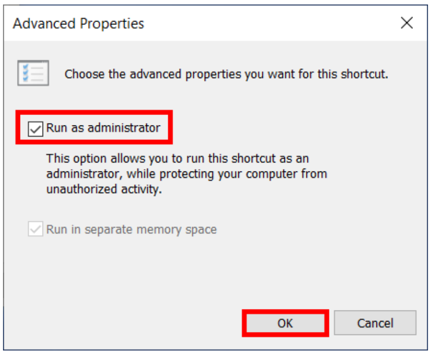
Target: "C:\Program Files\RA\RA.exe"

Start in: "C:\Program Files\RA\"

Shortcut key: None

Run: Normal window

Comment:



Advanced Properties

Choose the advanced properties you want for this shortcut.

☑ Run as administrator

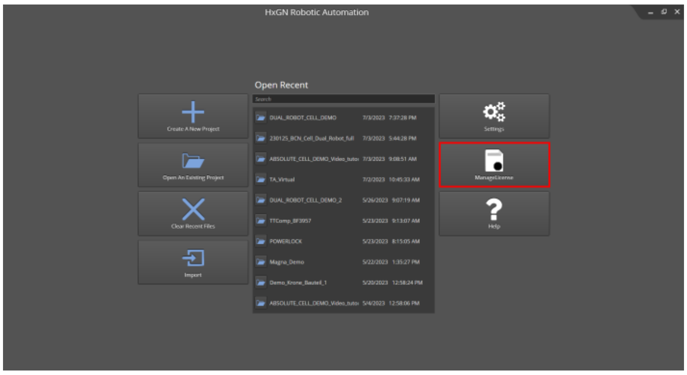
This option allows you to run this shortcut as an administrator, while protecting your computer from unauthorized activity.

☑ Run in separate memory space

3.2 Licensing

If this is the first installation, HxGN Robotic Automation will be launched in Viewer mode. Viewer mode will only allow you to open already created projects but will not allow you to execute programs and save projects.

To activate a license, from the welcome dashboard click on **Manage License** button and then click on **Activate** button.

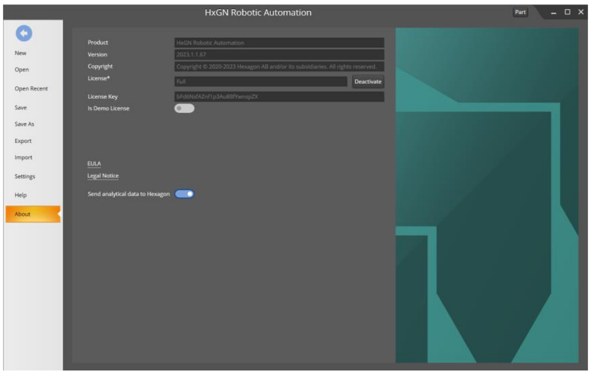


HxGN Robotic Automation

Open Recent

Search

DUAL_ROBOT_CELL_DEMO	7/3/2023	7:37:28 PM
230125_BCN_Cell_Dual_Robot_full	7/3/2023	5:44:28 PM
ABSOLUTE_CELL_DEMO_Video_tutor	7/3/2023	9:08:51 AM
TA_Virtual	7/2/2023	10:45:33 AM
DUAL_ROBOT_CELL_DEMO_2	5/26/2023	9:07:19 AM
TTComp_BF3957	5/23/2023	9:13:07 AM
POWERLOCK	5/23/2023	8:15:05 AM
Magna_Demo	5/22/2023	1:35:27 PM
Demo_Krone_Bauteil_1	5/20/2023	12:58:24 PM
ABSOLUTE_CELL_DEMO_Video_tutor	5/4/2023	12:58:06 PM



HxGN Robotic Automation

Product: HxGN Robotic Automation

Version: 2023.1.1.67

Copyright: Copyright © 2020-2023 Hexagon AB and/or its subsidiaries. All rights reserved.

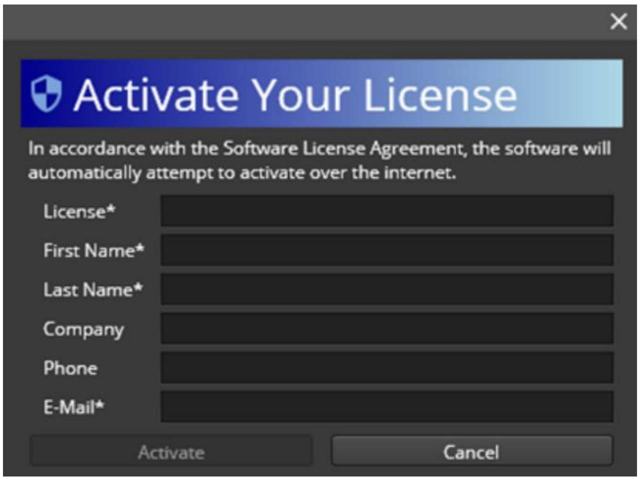
License*: Full

License Key: bFd6NsfAZnf1p3AuBBfYwmpZX

Is Demo License

Send analytical data to Hexagon

The **Activate Your License** dialog will appear. The user can enter the License key, fill the required fields, and click on **Activate** button.



Activate Your License

In accordance with the Software License Agreement, the software will automatically attempt to activate over the internet.

License*

First Name*

Last Name*

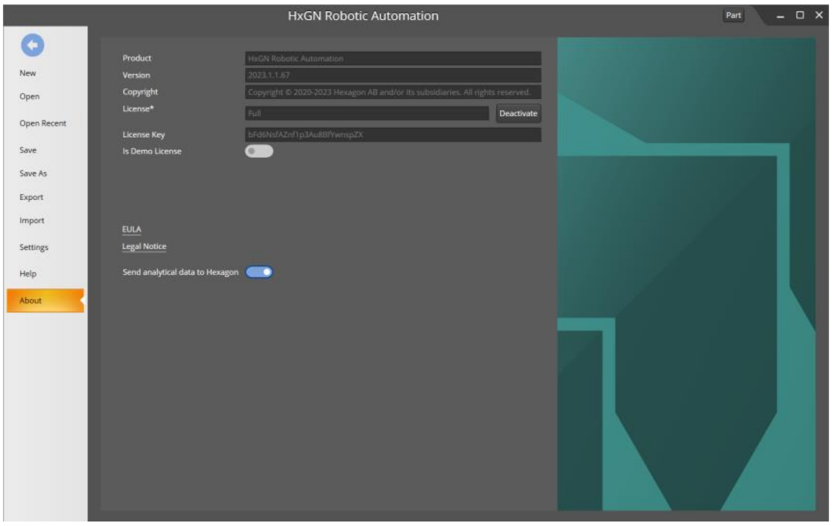
Company

Phone

E-Mail*

License activation is not allowed from Remote Session. The Software will automatically attempt to activate over the internet.

If the license key and all the other fields are correct, the license will be shown, and **Deactivate** button will appear.



HxGN Robotic Automation

Product: HxGN Robotic Automation

Version: 2023.1.1.67

Copyright: Copyright © 2020-2023 Hexagon AB and/or its subsidiaries. All rights reserved.

License*: Full

License Key: bFd6NsfAZnf1p3AuBBfYwmpZX

Is Demo License

Send analytical data to Hexagon

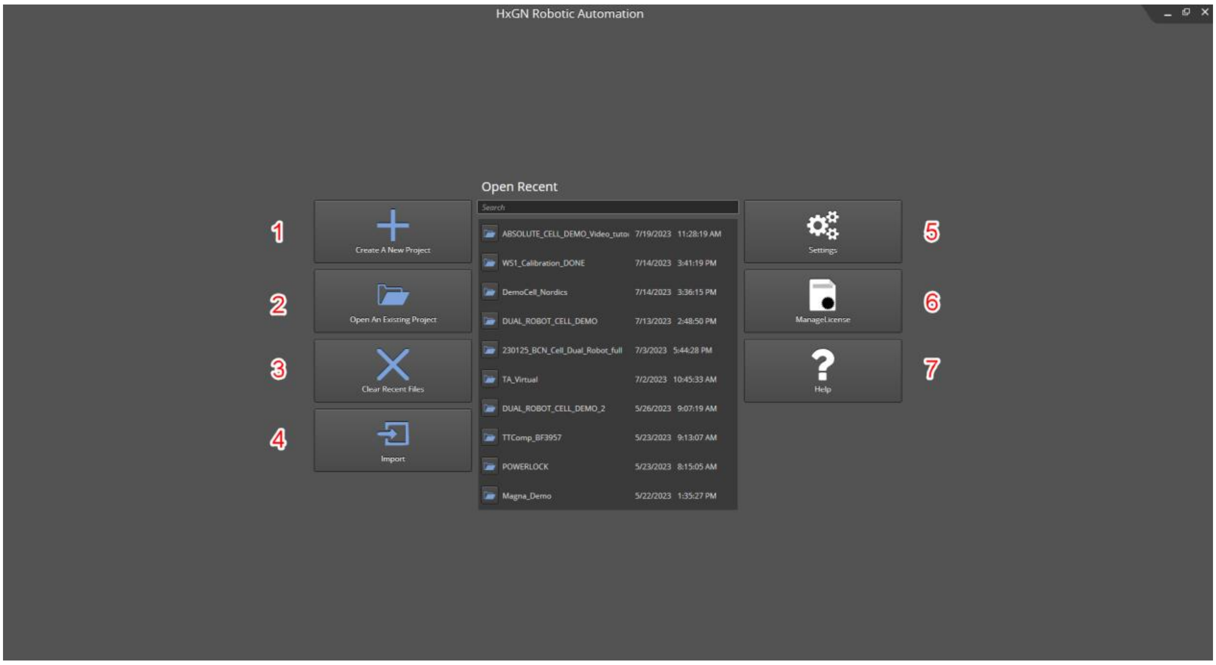
4 UI (User Interface) Overview

After launching the HxGN Robotic Automation application, the user can click on:

1) Create a new project to start building a new project from scratch

Warning: the project name must not contain any "." characters (e.g. "MyProject" is a valid name but "MyProject.2023" is not).

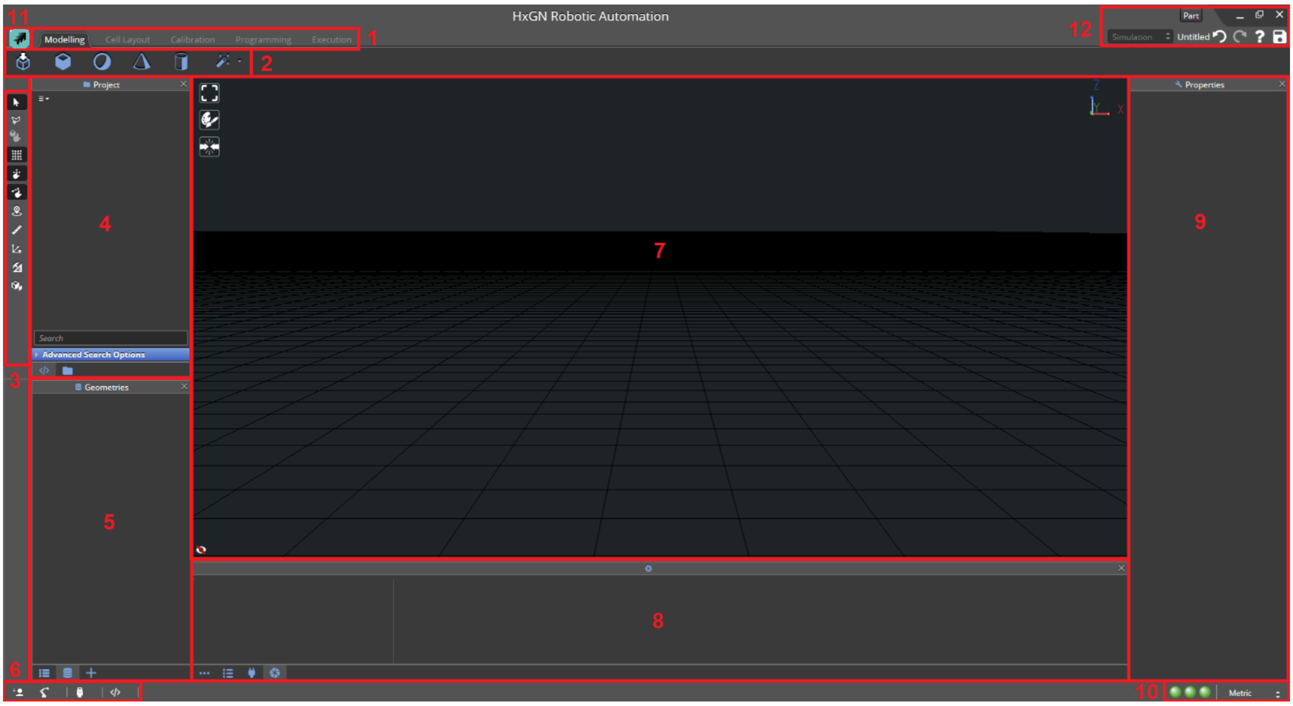
2) **Open and existing project**. The **Open Recent** area can also be used to choose a project that was recently used and a ***Search*** bar helps to filter among them.
3) **Clear recent files**, to remove all files from Open Recent area.
4) **Import**, to extract and open a project from a zip file previously exported from HxGN Robotic Automation (see chapter 4.11).
5) **Settings**, to navigate to the HxGN Robotic Automation settings, described in detail in chapter 4.11.1.
6) **Manage license**, to manage the activation of the license as explained in chapter 3.2.
7) **Help**, to be redirected to the web help.



ABSOLUTE_CELL_DEMO_Video_tuto	7/19/2023	11:28:19 AM
WS1_Calibration_DONE	7/14/2023	3:41:19 PM
DemoCell_Nordics	7/14/2023	3:36:15 PM
DUAL_ROBOT_CELL_DEMO	7/13/2023	2:48:50 PM
230125_BCN_Cell_Dual_Robot_full	7/3/2023	5:44:28 PM
TA_Virtual	7/2/2023	10:45:33 AM
DUAL_ROBOT_CELL_DEMO_2	5/26/2023	9:07:19 AM
TTComp_BF3957	5/23/2023	9:13:07 AM
POWERLOCK	5/23/2023	8:15:05 AM
Magna_Demo	5/22/2023	1:35:27 PM

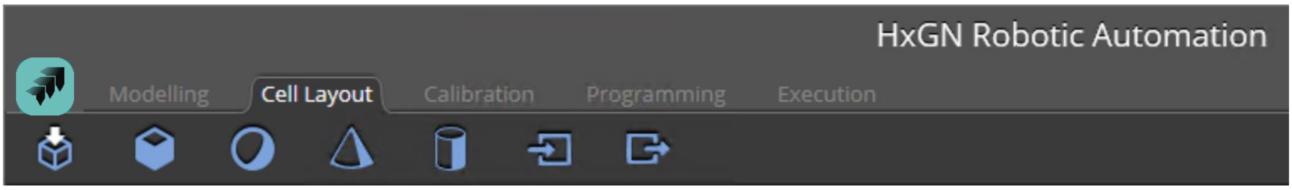
HxGN Robotic Automation UI can be divided into twelve fundamental areas.

1) Application Mode Tabs
2) Specific Application Mode Toolbar
3) Special Mode Toolbar
4) Project Tree and Library / Calibration / Program Windows
5) Geometries / Measurements / Conditions Windows
6) Active Machine / Active Sensor / Active Program Line Buttons
7) 3D Graphic Window
8) Status / Timeline / Connection / GoPro Windows
9) Properties Window
10) Metrology / Validation Engine / Collision Engine Buttons
11) RA Main Menu Button
12) Top-right Menu



4.1 Application Mode Tabs

HxGN Robotic Automation UI is mainly defined by five main tabs defining the Application modes, corresponding to the main user flows which are: Modelling, Cell Layout, Calibration, Programming and Execution.



4.1.1 Modelling

This is an application mode for advanced users. The focus is to enable the user to model any kinematic or static element. After modelling, you can export them to your library (right-click) and in this is the way generate your own elements catalogue to work with.

4.1.2 Cell Layout

The Cell Layout mode’s main purpose is to create a virtual representation of a cell with moving components. In this mode, the kinematic or static elements can be imported from the library or directly by importing CAD models and placed according to the cell layout of real or simulated cells. Existing cell layouts can be edited.

4.1.3 Calibration

Once the elements are positioned and oriented in the cell, the next step is going to be Calibration. Calibrating a cell means aligning the elements in the virtual environment exactly according to where the cell elements are in the real physical cell. This step is critical for the process since any difference between real and virtual cells will affect path planning, collision detection, scans that will be received from the robot, etc.

This mode has two categories of wizards:

1) Calibration Wizard: Robot Calibration wizards.
2) Alignment Wizard: Floor Alignment, Cell Alignment, Part Alignment, Best Fit Alignment wizards or Add Reflectors Wizard.

The first step is to align all the elements in the cell, using the Floor, Cell and Part alignment. Then with the cell aligned, the robot and the tool must be calibrated to have them placed as they are in the real cell in the most precise way.
Lastly you can Add Reflectors through the wizard that are either linked to the Turn Table (and will rotate with it) or placed at fixed location within the cell. Reflectors can either be added through picking on CAD (selecting it in the cell UI) or through measuring them with the tracker.

4.1.4 Programming

In this application mode, the user can:

1) Create Programs in Manual or Automatic Mode for Part Inspection.
2) Edit existing programs e.g., to adjust them to be optimized for the required task the robot will perform.
3) Validate the programs to be collision free and evaluate the execution time using the Timeline.
4) Set the program specific properties (define the Metrology application to be used).
5) Get the features from the connected Metrology application.

Depending on the application the following could be considered:

- If the program will be for measuring parts and executing it online (not simulation), make sure your cell is aligned, the robot and tool are calibrated, your fixture, and part are loaded, properly aligned and the features you are planning on measuring are imported (from Robotic Automation or Metrology).
- If the program will be for measuring parts and execute it online, make sure your cell is aligned, robot and tool are calibrated, you have the Turn Table Compensation map in case there is Turn Table in the project, your fixture and part are loaded, properly aligned and the features you are planning on measuring are imported.

4.1.5 Execution

Application mode where the user can:

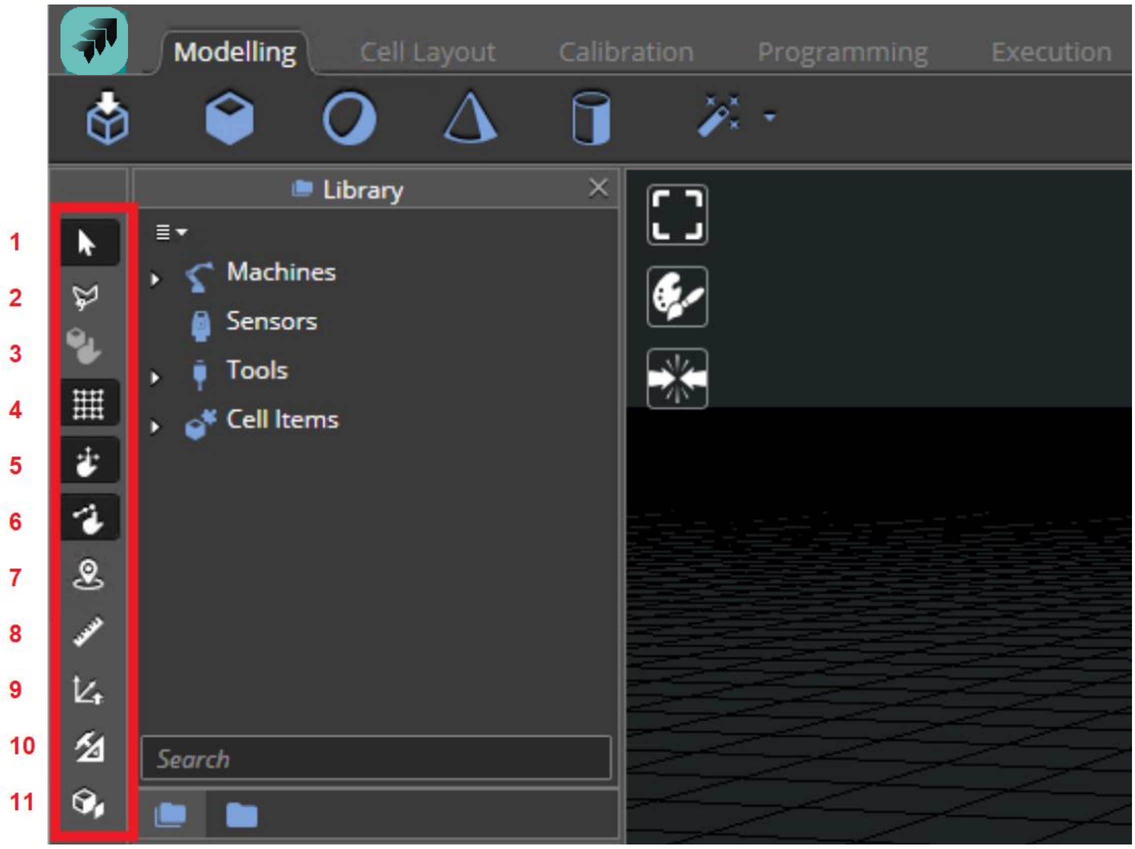
- Execute Programs Offline to simulate the inspection process and see the simulated data.
- Execute Programs Online to collect actual data that is sent to Metrology.
- Control the Program Execution (start, resume, stop)
- Export programs to be executed by Execution Manager.

4.2 Specific Application Mode Toolbar

This toolbar changes depending on the Application Mode in use, and it will be explained in detail in the dedicated chapters.

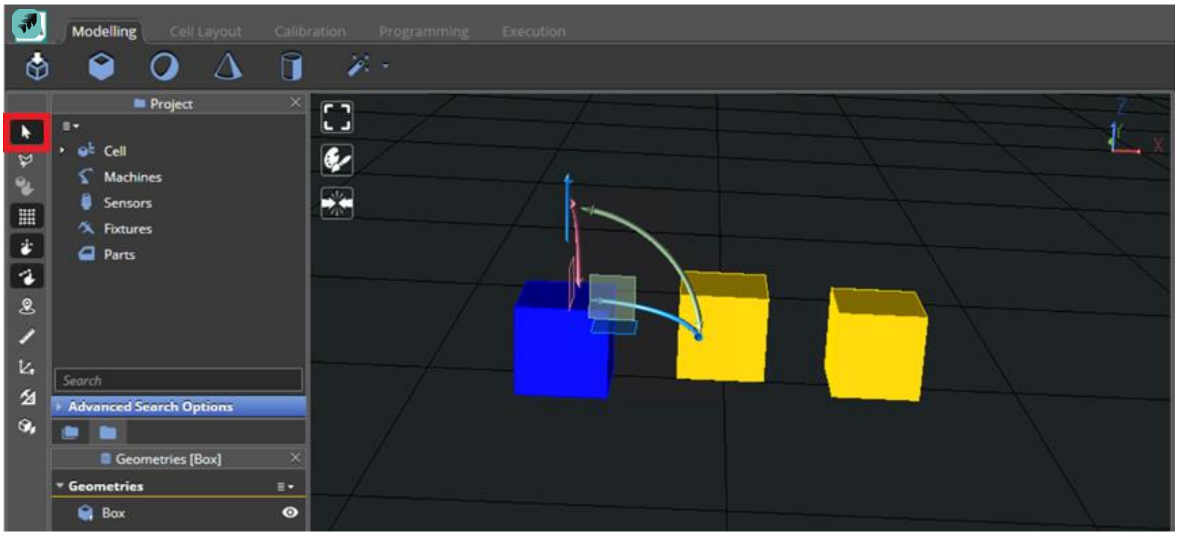
4.3 Special Mode Toolbar

Special modes toolbar is found on the left side of the window.



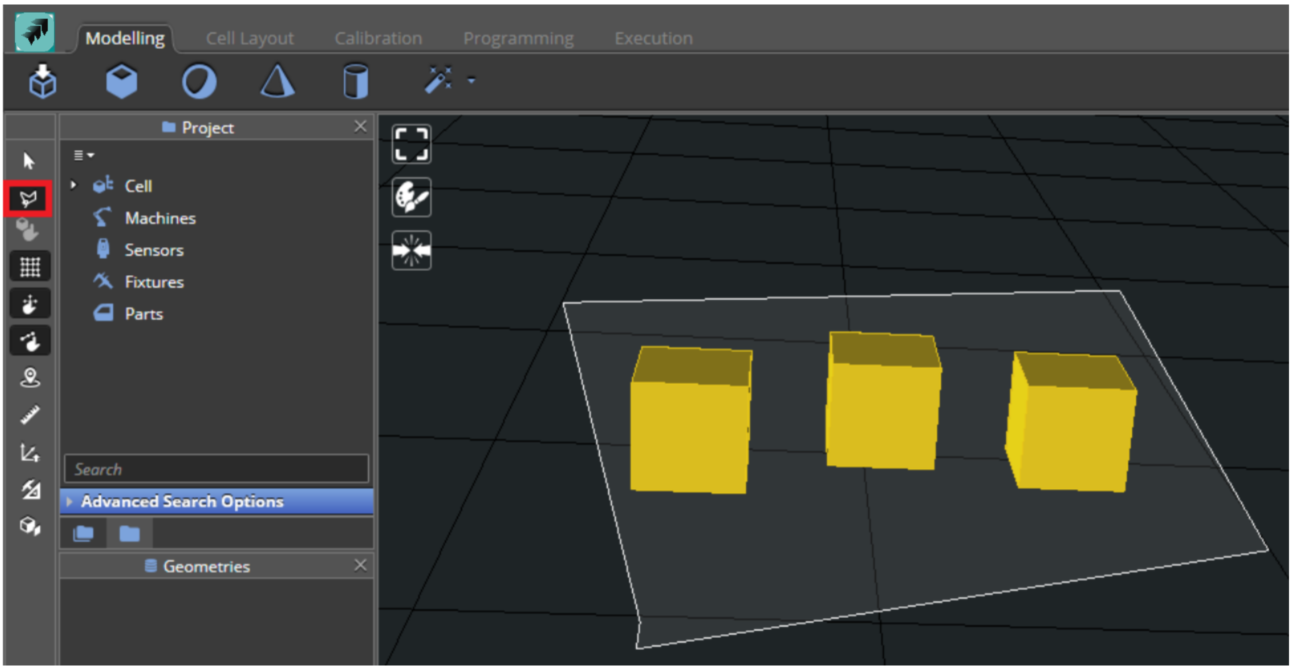
4.3.1 Regular Selection

Regular selection / polygonal selection controls the way elements are selected on Robotic Automation. As you can see in the image below, regular selection makes the mouse cursor able to select a single object.



4.3.2 Polygonal Selection

Polygonal selection makes the mouse cursor to be able to draw a polygonal area where cell elements are present. Do the selection with the left button of the mouse and do right-click when the desired polygon is created, then all the cell elements within the polygonal selection will be selected and highlighted after pressing the [Enter] key.

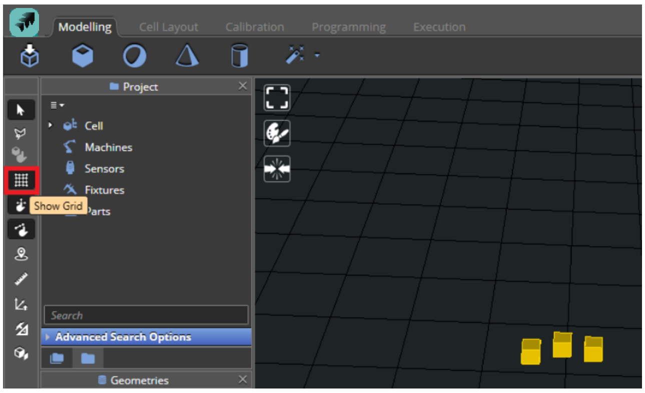


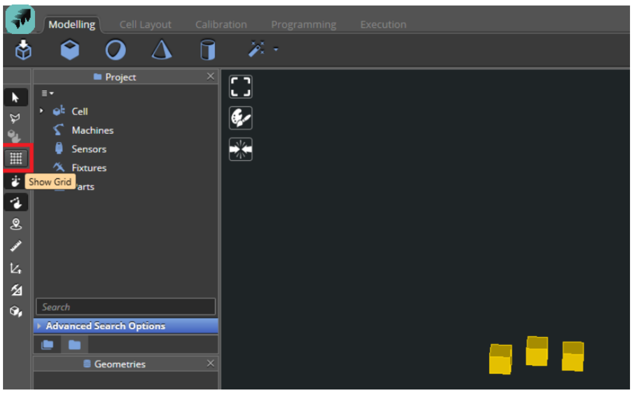
4.3.3 Pick on CAD

Pick on CAD is a special mode, used to adjust robot position using the TCP (Tool Center Point) or directly create manually robot positions. It can be only activated in programming mode, see chapter 8.10.4.

4.3.4 Show/hide grid

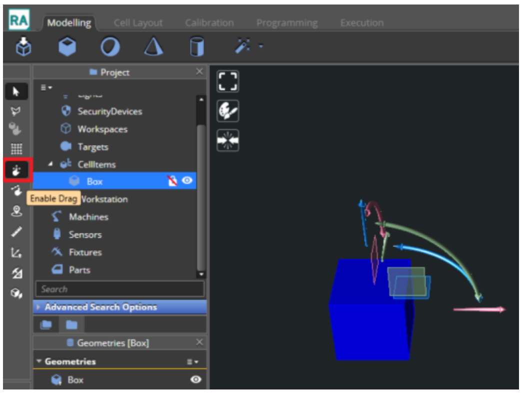
Toggle the “Show/hide grid” button to visualize or hide the grid in the graphic window.

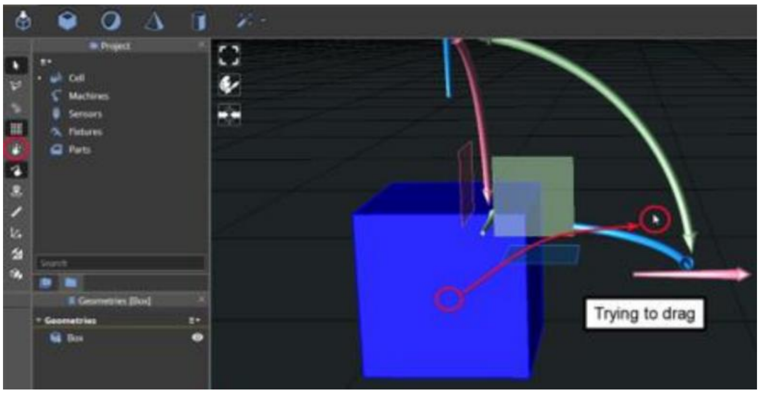




4.3.5 Enable/disable drag

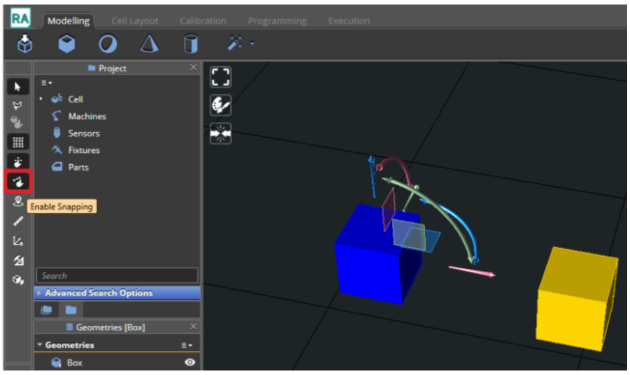
Toggle the “Enable/disable drag” button to be able to drag or to disable the dragging. If dragging is disabled, if the user tries to drag the object, it won’t move.

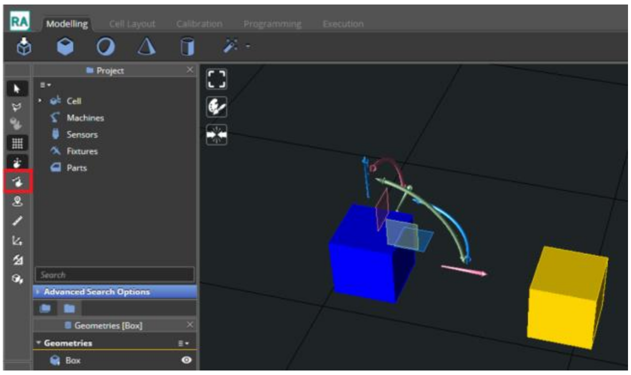




4.3.6 Enable/disable snapping

Toggle the “Enable/disable snapping” button to be able to snap or to disable snapping.



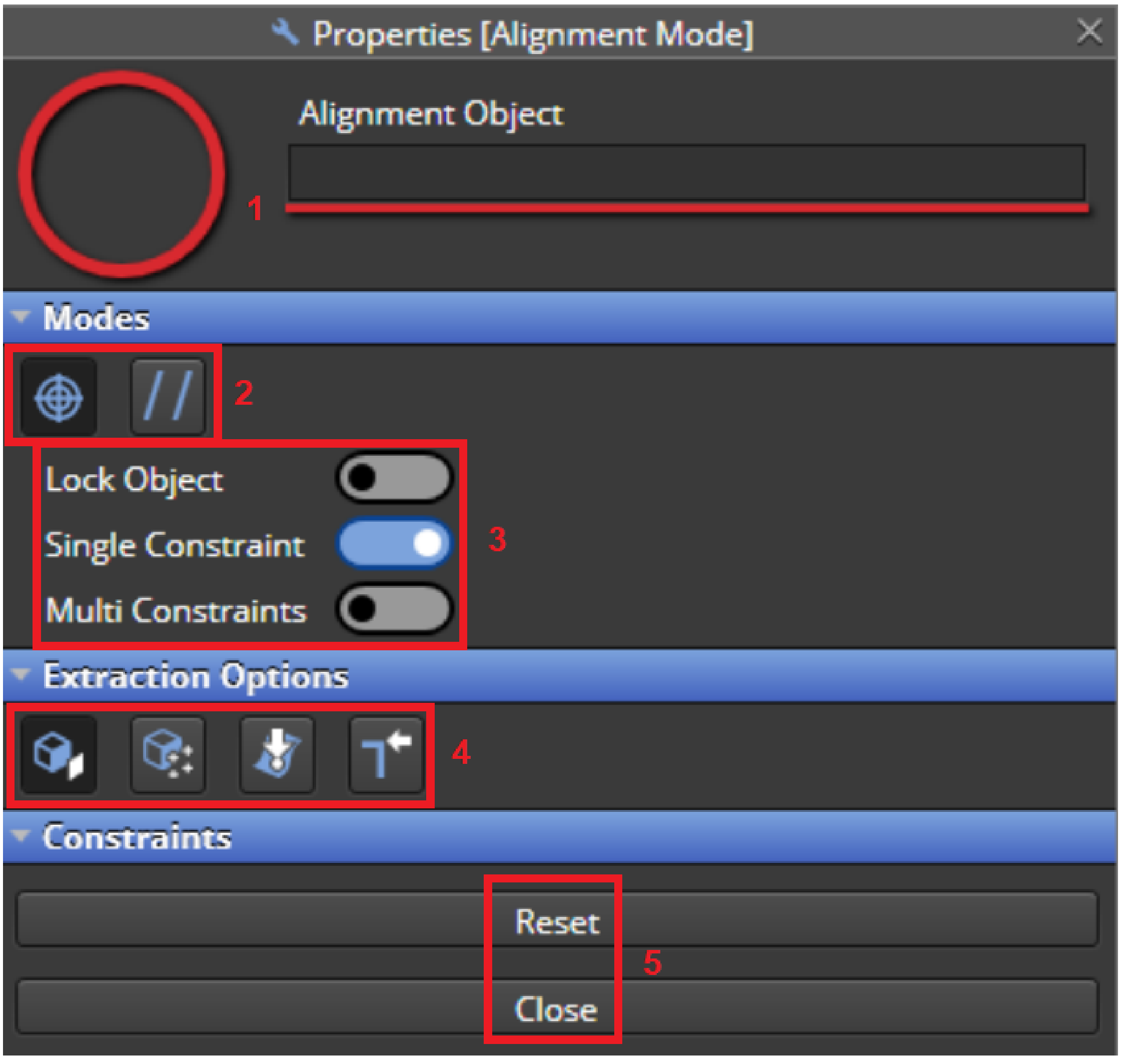


4.3.7 Alignment Mode

Alignment brings the possibility to the user to set up constraints between objects or features to align them.

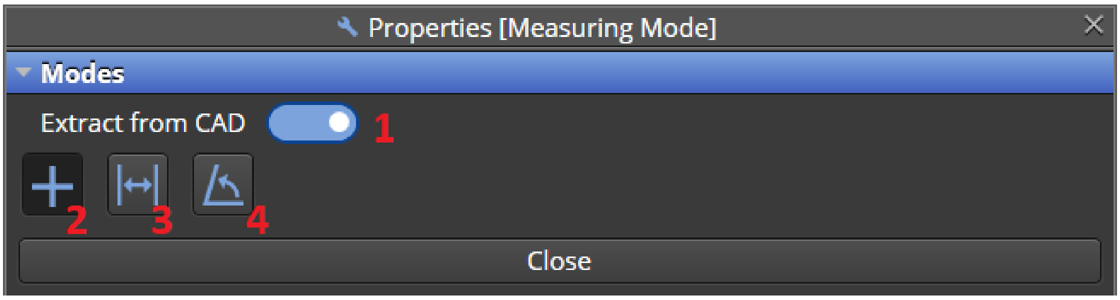
1) Name of the object that you are currently aligning.
2) Select the align mode: coaxial or parallel.
3) Selections switches:
    a) Lock Objects: if checked, the object will be locked after finishing the alignment.
    b) Single Constraints: if checked, the single constraint mode will be selected.
    c) Multi Constraints: if checked, the multiple constraint mode will be selected.
4) Extraction features modes:
    a) Feature: extracts feature from object.
    b) Feature from points: extracts feature generated from points.
    c) Surface point: extracts point from surface.
    d) Edge point: extracts points from edge
5) Reset or close the current alignment.

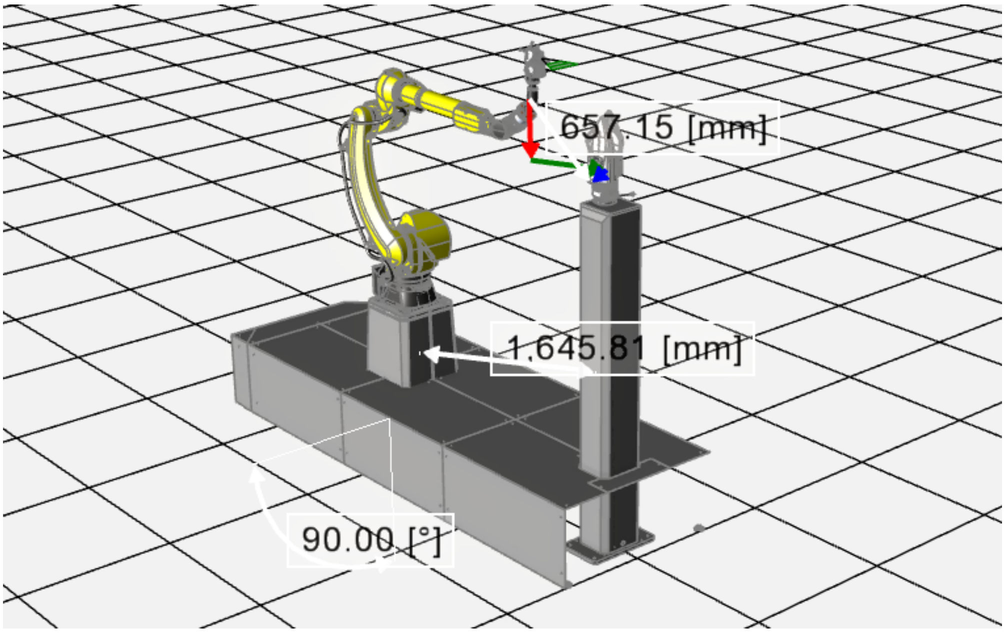
For more details on how to use the manual alignment mode, please refer to chapter 7.1



4.3.8 Measuring Mode

1) **Extract from CAD**: if checked the features that you will measure will be from the CAD.
2) **Free Point**: to calculate the distance between two free points selected in the cell.
3) **Distance**: to calculate the distance between two surfaces selected in the cell.
4) **Angle**: to calculate the angle between two surfaces selected in the cell.

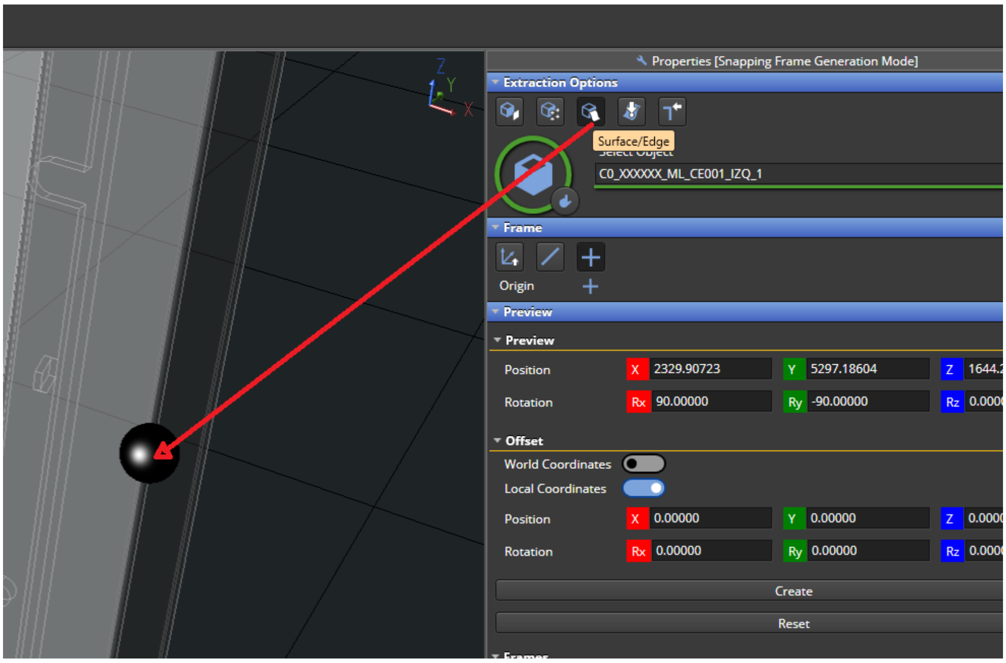




4.3.9 Snapping Frame Generation Mode (only available in Modelling mode)

This mode is to generate frames to an object. Please refer to chapter 5 for more information about its usage in Modelling mode. A snapping frame is a point added to a cell item, once 2 objects with snapping frames get closer, a bubble appears and if we bring them close enough and stop dragging, the items will be aligned by the snapping frame.
There are few options to extract features:

1) Feature: extracts feature from object.
2) Feature from points: extracts feature generated from points.
3) Surface/Edge: This is to extract the point from Edge between two surfaces:



Select Object: C0_XXXXXX_ML_CE001_IZQ_1

Preview

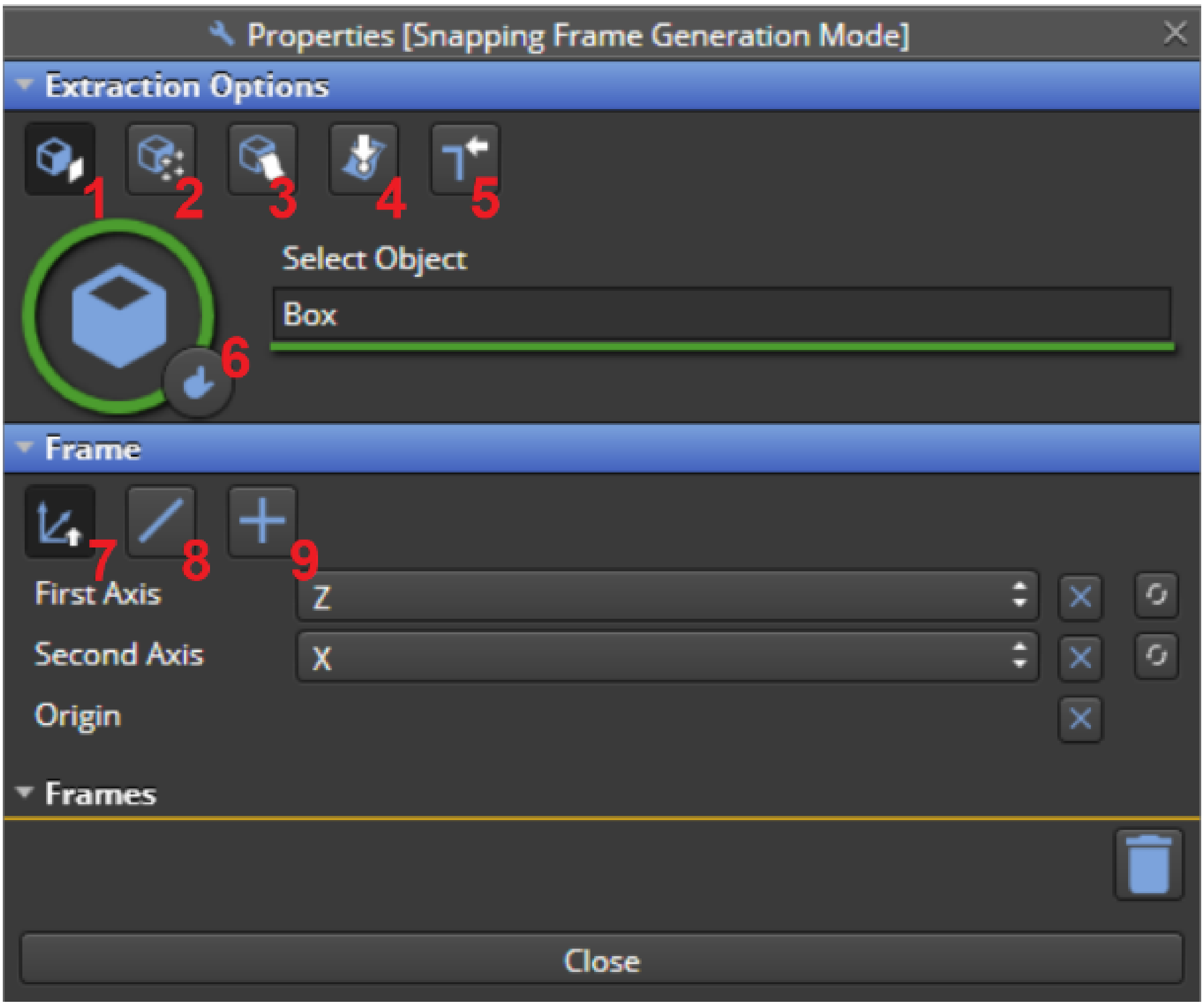
Position	X 2329.90723	Y 5297.18604	Z 1644.2
Rotation	Rx 90.00000	Ry -90.00000	Rz 0.0000

Offset

World Coordinates ○
Local Coordinates ●

Position	X 0.00000	Y 0.00000	Z 0.0000
Rotation	Rx 0.00000	Ry 0.00000	Rz 0.0000

4) Surface point: extracts point from surface.
5) Edge point: extracts points from edge.
6) Select the object that you want to create a snapping frame on.
7) Pick first axis of the frame then pick the second axis of the frame and to finish pick the origin of the frame.
8) Pick the origin of the normal and then pick the origin of the normal again.
9) Pick the origin of the point.



Properties [Snapping Frame Generation Mode]

Extraction Options

Select Object: Box

Frame

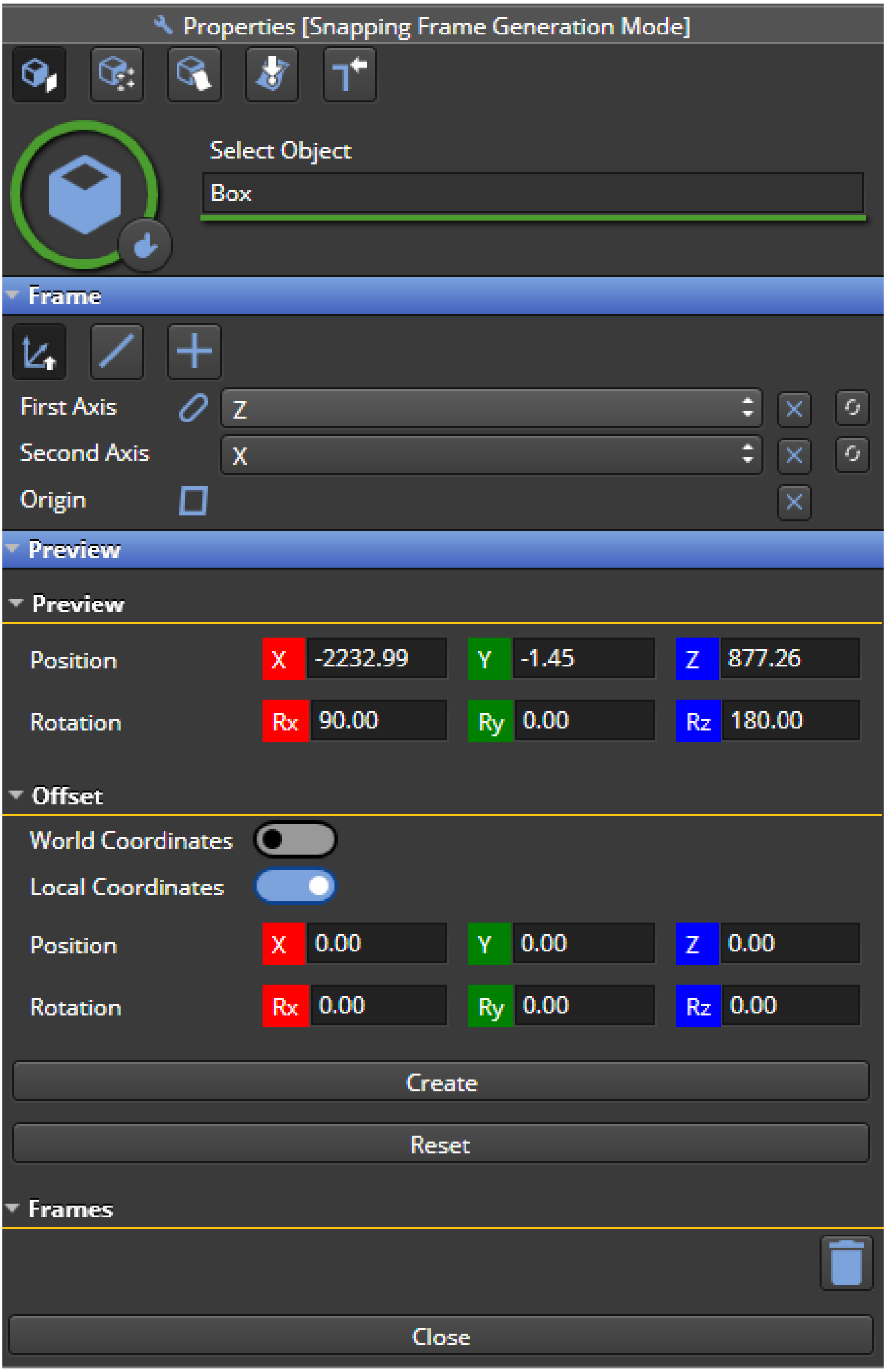
First Axis: Z

Second Axis: X

Origin

Frames

Once the object and the frames conditions are selected, we can see a preview of the position and rotation of the frame. Also, we can choose into “World Coordinates” or Local Coordinates”. Then we can press “Create” to create it or “Reset” to reset it.



Properties [Snapping Frame Generation Mode]

Select Object: Box

Frame

First Axis: Z

Second Axis: X

Origin

Preview

Preview

Position: X -2232.99 Y -1.45 Z 877.26

Rotation: Rx 90.00 Ry 0.00 Rz 180.00

Offset

World Coordinates: off

Local Coordinates: on

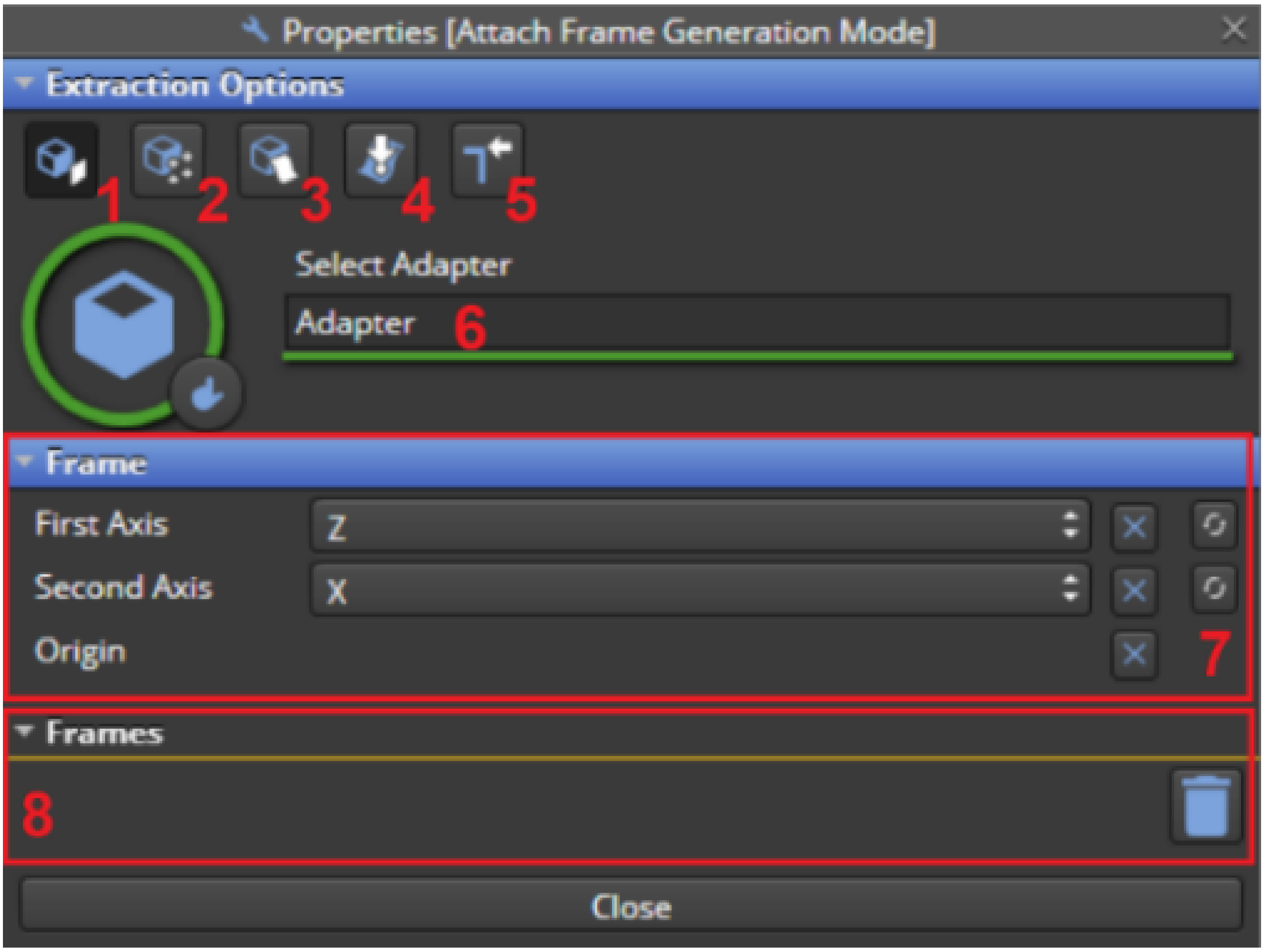
Position: X 0.00 Y 0.00 Z 0.00

Rotation: Rx 0.00 Ry 0.00 Rz 0.00

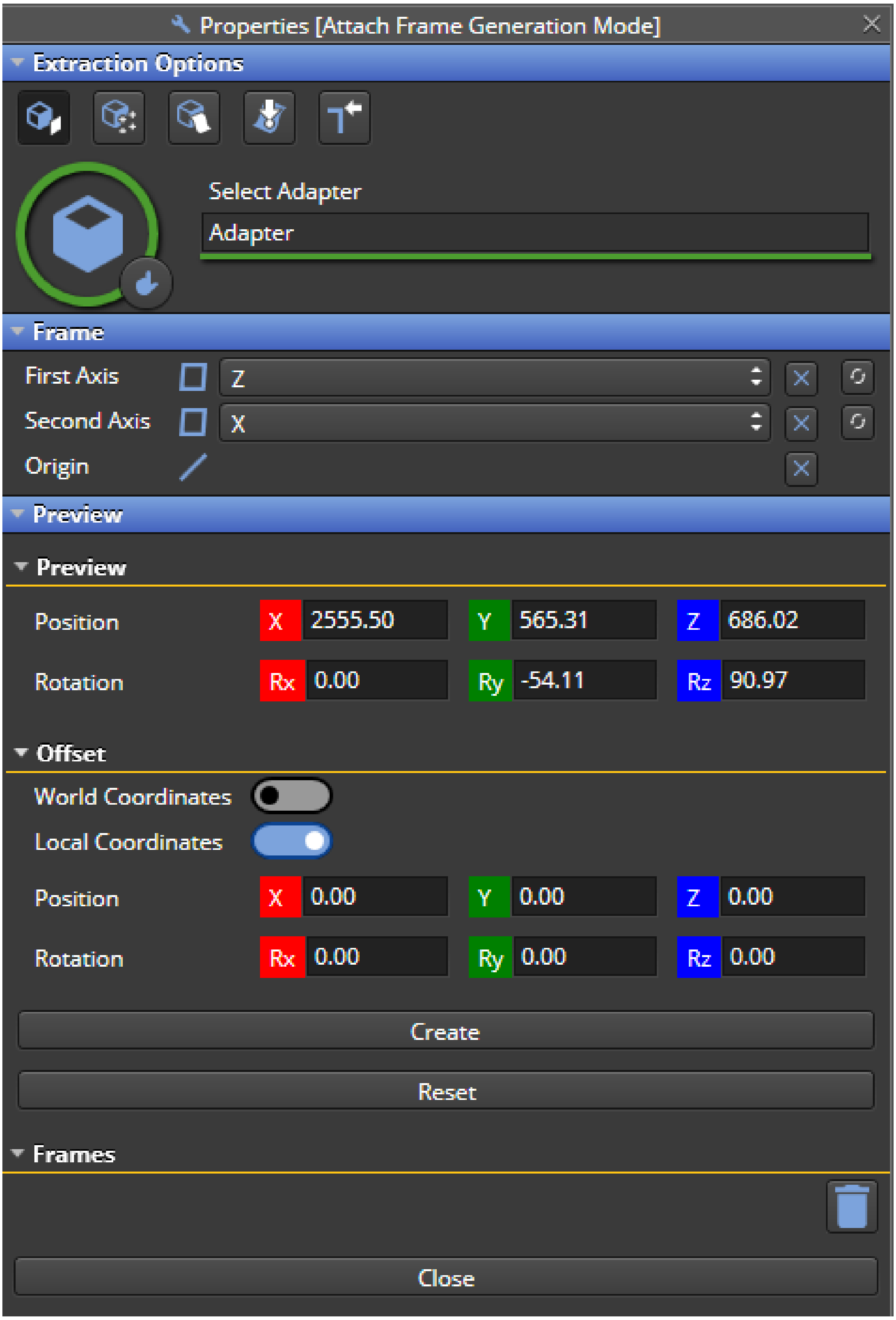
4.3.10 Attach Frame Generation Mode (only available in Modelling mode)

This mode is to generate frames to an adapter. Please refer to chapter 5 for more information about its usage in Modelling mode.

1) Extraction features mode.
2) Feature: extracts feature from object.
3) Feature from points: extracts feature generated from points.
4) Surface point: extracts point from surface.
5) Edge point: extracts points from edge.
6) Select the adapter that you want to create attaching frame on.
7) Select the axis and the origin of the frame.
8) List of the frames created on the adapter.



Once the adapter is selected and the axis and the origin are picked, we can see a preview of the position and rotation of the frame. Then if you want to create the frame you must click on "Create" and then the frame will be created and will appear on the list. We can reset as well if the preview is not how we expected and try again.



Properties [Attach Frame Generation Mode]

Extraction Options

Select Adapter: Adapter

Frame

First Axis: Z

Second Axis: X

Origin

Preview

Preview

Position	X 2555.50	Y 565.31	Z 686.02
Rotation	Rx 0.00	Ry -54.11	Rz 90.97

Offset

World Coordinates ○

Local Coordinates ●

Position	X 0.00	Y 0.00	Z 0.00
Rotation	Rx 0.00	Ry 0.00	Rz 0.00

Create

Reset

Frames

Close

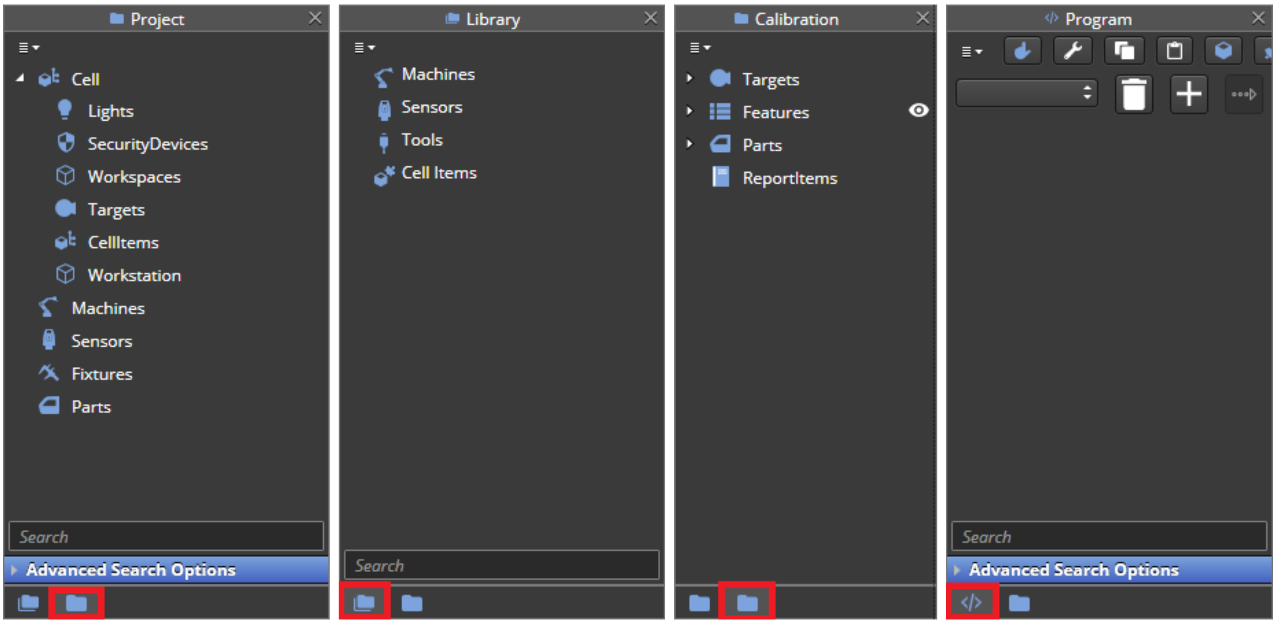
4.3.11 Feature Generation Mode

Click the button to open the Feature Generation Mode, see chapter 6.2 for more information.

4.4 Project Tree / Library / Calibration / Program Windows

This window is divided into two tabs depending on the specific application mode used.

- If using either Modelling or Cell Layout modes, the **Library** and **Project Tree** tabs will be shown
- in case Calibration mode is used, the **Project Tree** and **Calibration** tabs
- using either Programming or Execution modes, the **Program** and **Project Tree** tabs.



The **Project** tree is defined by five areas:

1) **Cell**: in this list, you can find:
   a) **Lights**: currently not supported
   b) **Security devices**: currently not supported
   c) **Workspaces**: Any workspace the user created will be displayed here.
   d) **Targets**: Any reflectors created either manually or through the TT compensation will be displayed here, as well as T-Probes.
   e) **Cell items**: All the items in the cell will be displayed here (walls, fences, and other physical objects).
   f) **Workstation**: defines the cell elements used by the project as defined in the Cell.json configuration file (see HxGN Robotic Automation Service Installation Manual for a detailed explanation).
2) **Machines**: On this list, you will find all the machines this project has (Robots, Sliders, Turntables).
3) **Sensors**: All available sensors (for example Trackers) will be displayed under this tree.
4) **Fixtures**: All fixtures in the cell will be displayed here.
5) **Parts**: All the parts in this cell, including their features, will appear under this tree.

Similarly, the **Library** is composed by four areas with the same definitions supplied above for the project tree:

1) Machines
2) Sensors
3) Tools
4) Cell Items

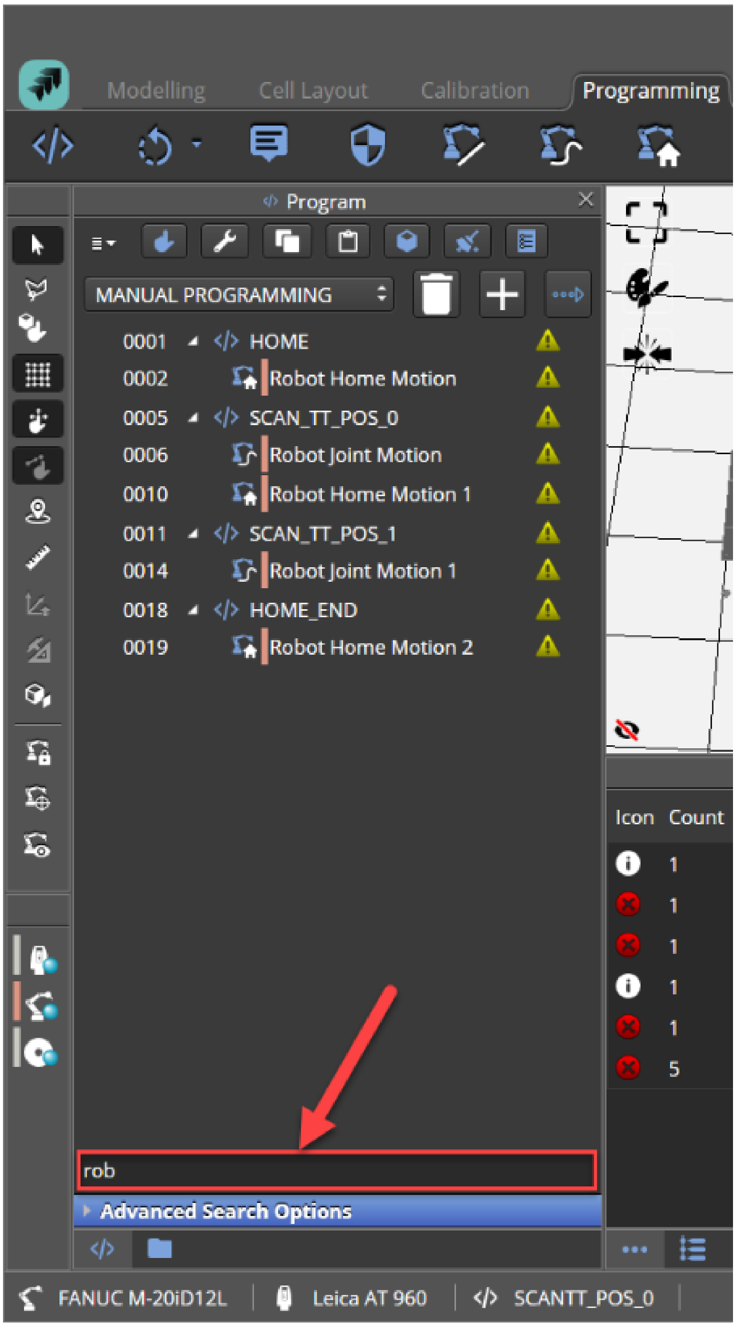
For details on the **Calibration** window please refer to chapter 7.

For details on the **Program** window please refer to chapter 8.

Pro Tip: by selecting any item in the Project tree and using the [CTRL + F] keyboard shortcut, the focus will be jumping to the selected item in the 3D Window.

4.4.1 Search bar and Advanced Search Options

Project, Program and Library windows have a search bar where the user can write the text to be search within the project tree, library items or program commands respectively.

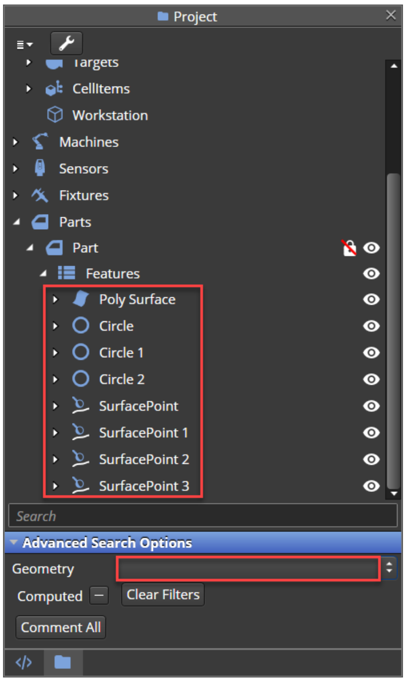


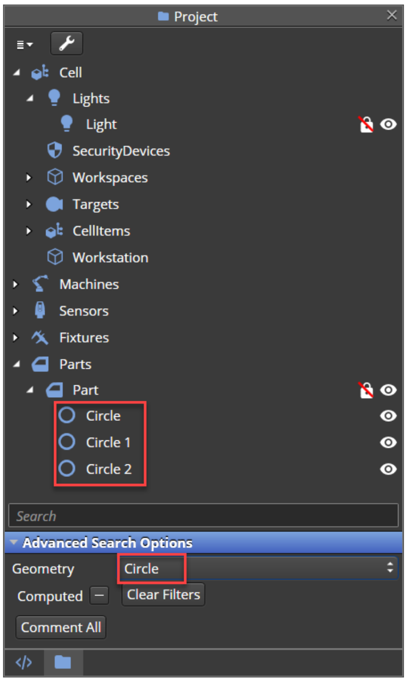
```
0001  HOME
0002      Robot Home Motion
0005  SCAN_TT_POS_0
0006      Robot Joint Motion
0010      Robot Home Motion 1
0011  SCAN_TT_POS_1
0014      Robot Joint Motion 1
0018  HOME_END
0019      Robot Home Motion 2
```

rob

By clicking on **Advanced Search Options** it is possible to show additional capabilities.

In Project window it is possible to filter the features by the geometry types available within the project.

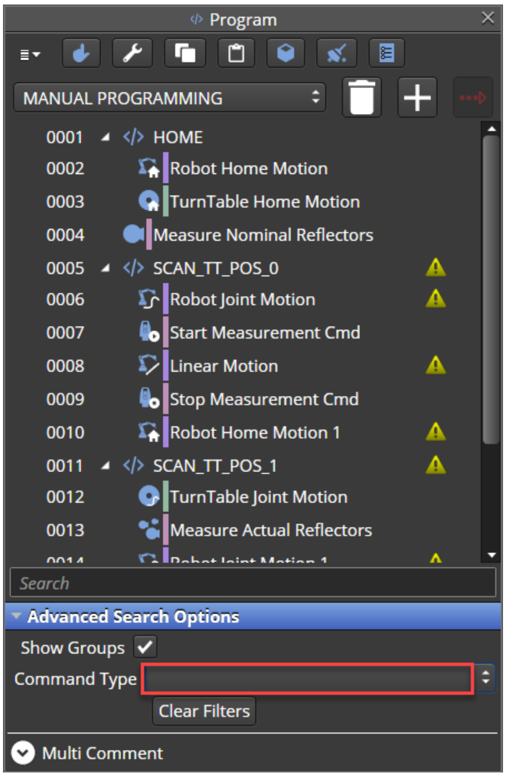


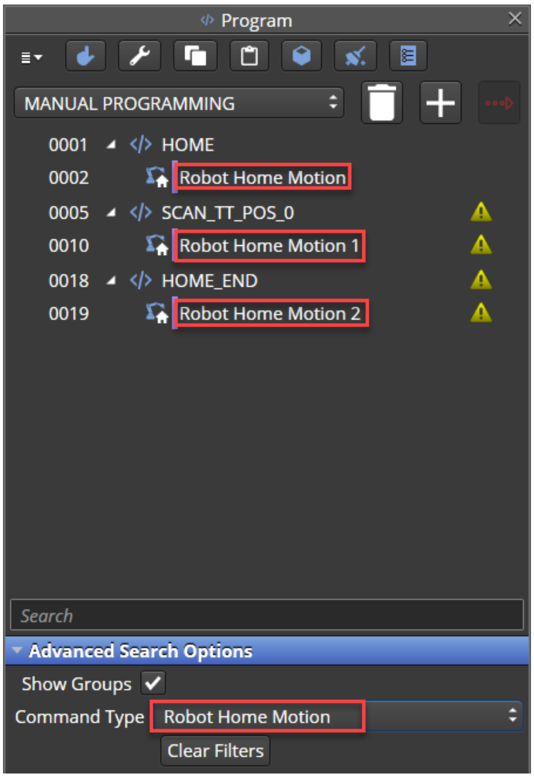


Geometry: Circle

In Program window is possible to:

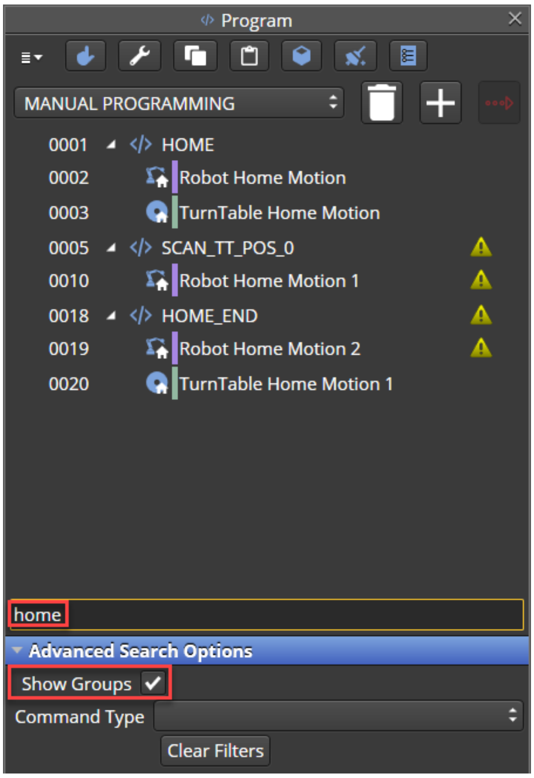
1) filter command by type selecting it in the **Command Type** drop-down menu:





Command Type: Robot Home Motion

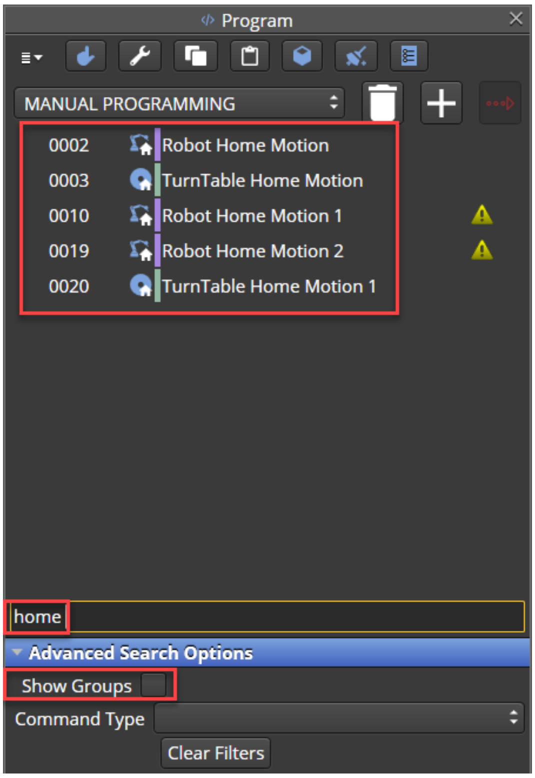
2) show or hide the command groups after searching for specific text with the search bar by checking the **Show Groups** checkbox:



0001 HOME
0002 Robot Home Motion
0003 TurnTable Home Motion
0005 SCAN_TT_POS_0
0010 Robot Home Motion 1
0018 HOME_END
0019 Robot Home Motion 2
0020 TurnTable Home Motion 1

home

Show Groups ☑

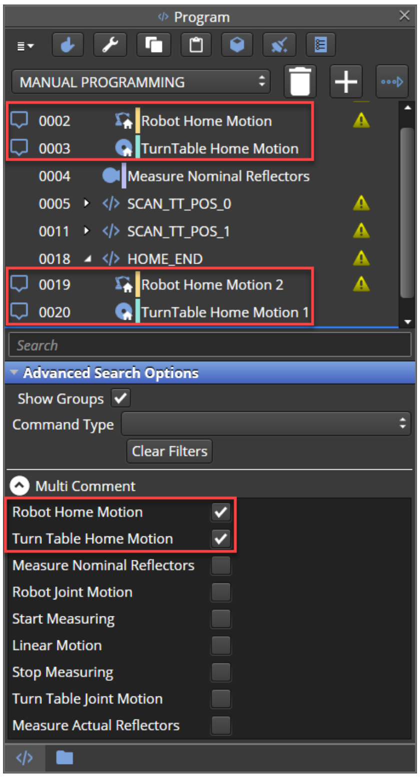


0002 Robot Home Motion
0003 TurnTable Home Motion
0010 Robot Home Motion 1
0019 Robot Home Motion 2
0020 TurnTable Home Motion 1

home

Show Groups □

3) perform a multi command comment for all the commands of the same type:



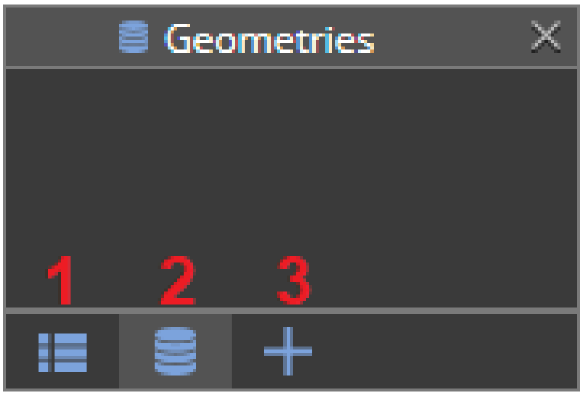
0002 Robot Home Motion
0003 TurnTable Home Motion
0004 Measure Nominal Reflectors
0005 SCAN_TT_POS_0
0011 SCAN_TT_POS_1
0018 HOME_END
0019 Robot Home Motion 2
0020 TurnTable Home Motion 1

Show Groups ☑

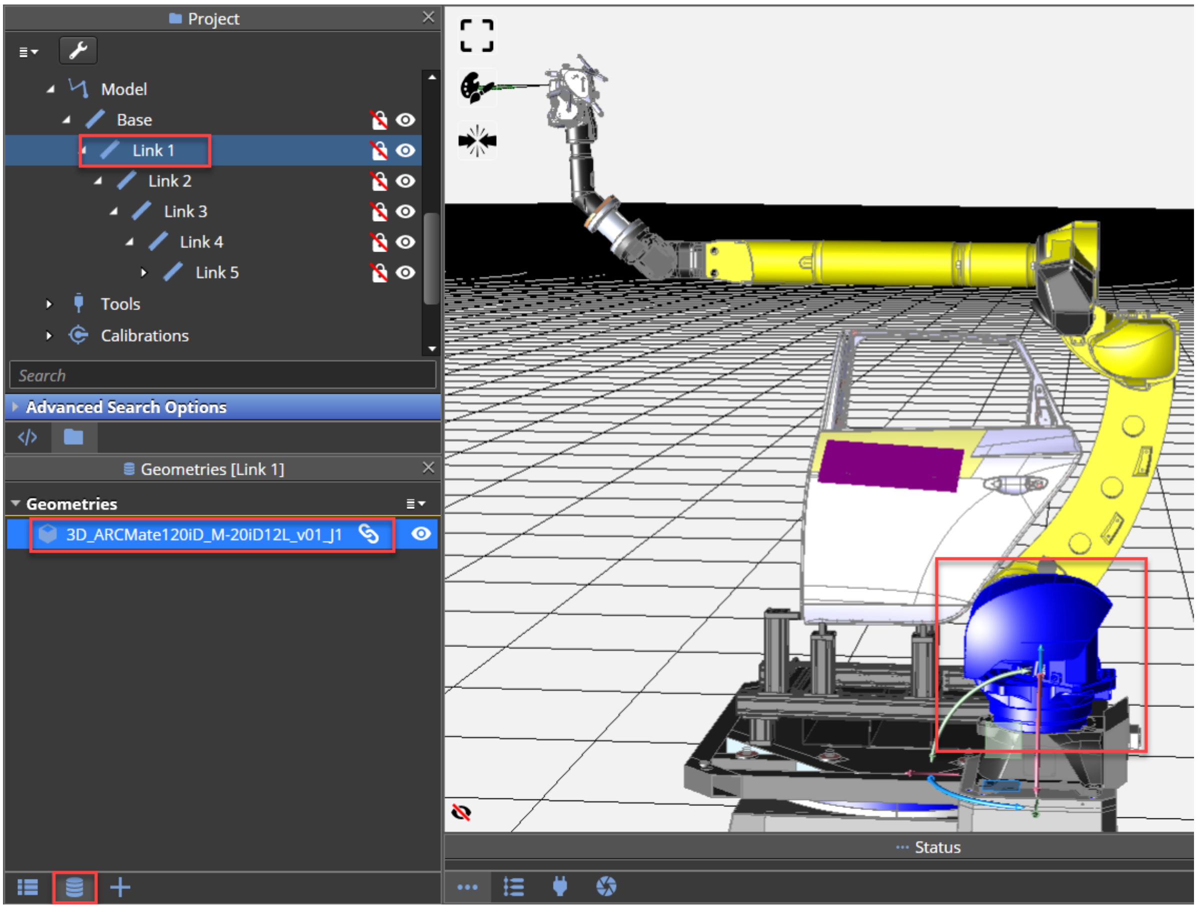
Robot Home Motion ☑
Turn Table Home Motion ☑
Measure Nominal Reflectors □
Robot Joint Motion □
Start Measuring □
Linear Motion □
Stop Measuring □
Turn Table Joint Motion □
Measure Actual Reflectors □

4.5 Geometries / Measurements / Conditions Windows

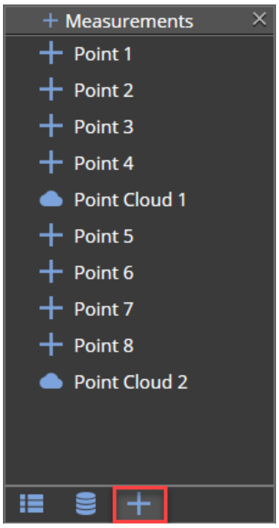
This window is made of three tabs:



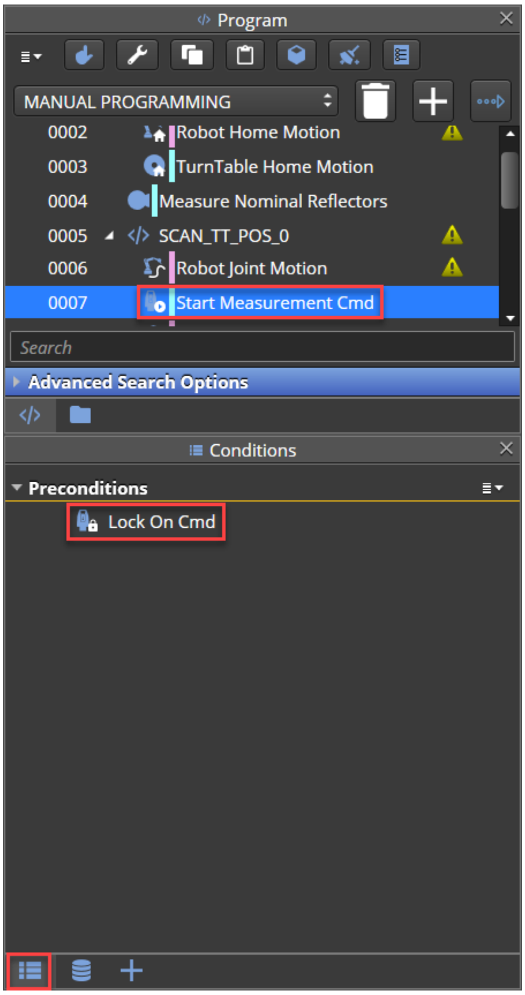
1) **Geometries** (always visible in all modes): this area contains the geometrical objects files that are associated with any items modelled within the cell (e.g. a robot joint CAD drawing).



2) **Measurements** (visible only in Calibration, Programming and Execution modes): this area will hold the points measured by the tracker and the COP (Cloud of Points) acquired by the laser scanner during program execution.

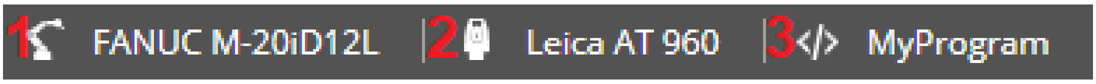


3) **Conditions** (visible only in Programming and Execution modes): this area shows the pre conditions required by some commands to be executed (e.g. a Start Measurement commands has a Lock On command as a pre-requisite, as the tracker needs first to lock on a scanner face for the scanner to be able to start scanning the part and collecting the COP).



4.6 Active Machine / Active Sensor / Active Program Line Buttons

This area in the lower-left corner of the UI shows the selected elements of the projects.

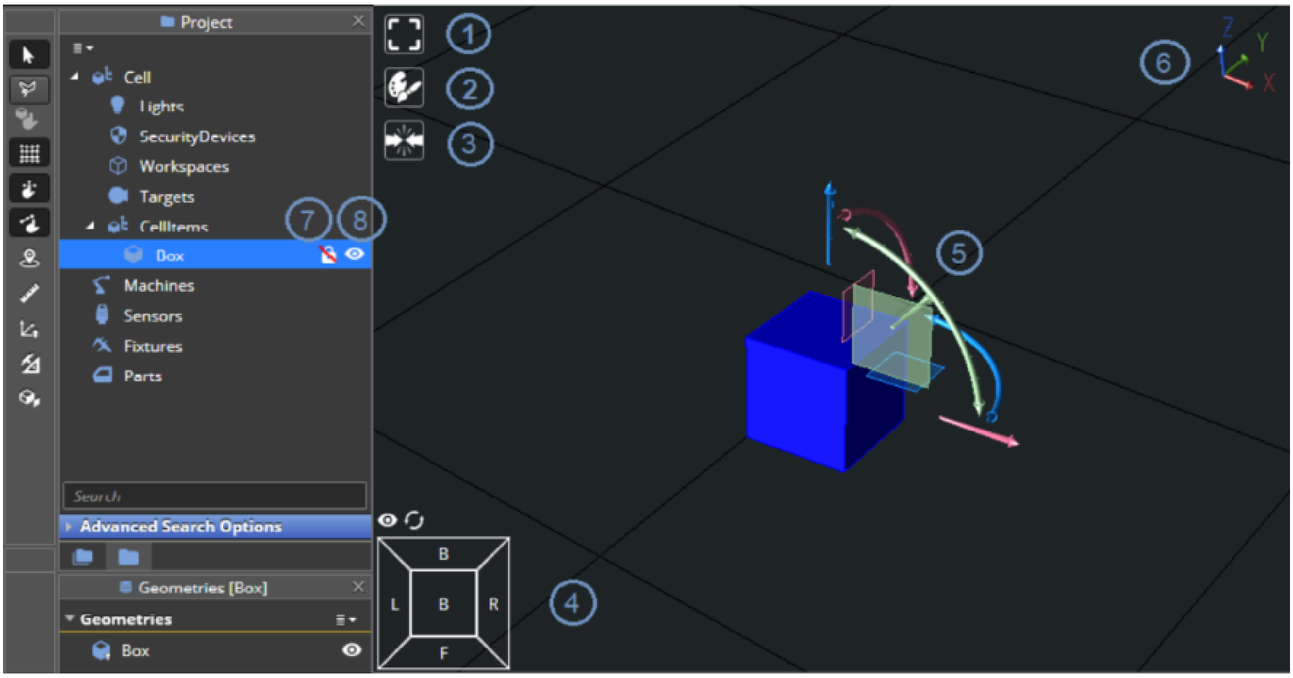


1) **Active Machine**: the machine currently selected, e.g., a robot or a rotary table
2) **Active Sensor**: the sensor currently selected, e.g., the Leica AT 960 tracker
3) **Active Command**: the program name (e.g., MyProgram) or program command currently selected

4.7 3D Graphic Window

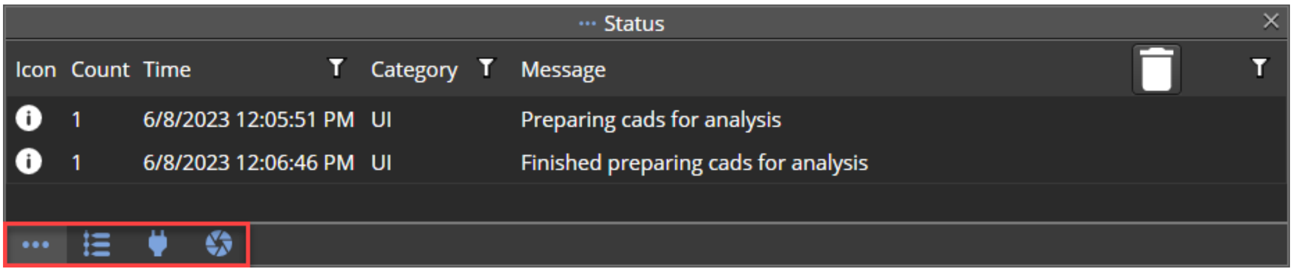
The Graphic Window can be divided into eight parts:

1) Fit all in view or Zoom in Selection
2) Graphic Menu
3) Collision Menu
4) View Modes
5) Manipulator to Drag and Rotate Selected Items
6) Cell Coordinate System
7) Object Position Lock/Unlock
8) Object Visibility Show/Hide



4.8 Status / Timeline / Connection / GoPro Windows

This window is divided into four main tabs as described below.

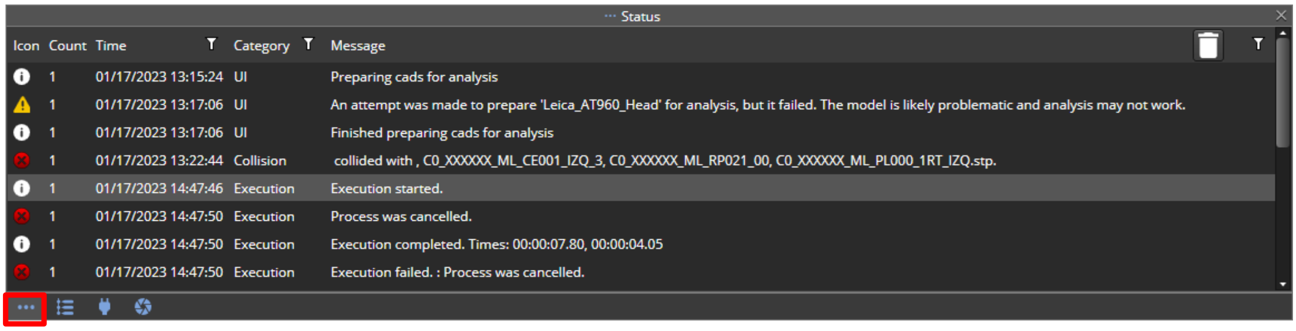


··· Status

Icon	Count	Time	Category	Message
ⓘ	1	6/8/2023 12:05:51 PM	UI	Preparing cads for analysis
ⓘ	1	6/8/2023 12:06:46 PM	UI	Finished preparing cads for analysis

4.8.1 Status

The Status window will show all the information, warning and error messages from the various user actions or software actions (errors in loading, CAD (Computer Aided Design) loaded, program started, etc.).

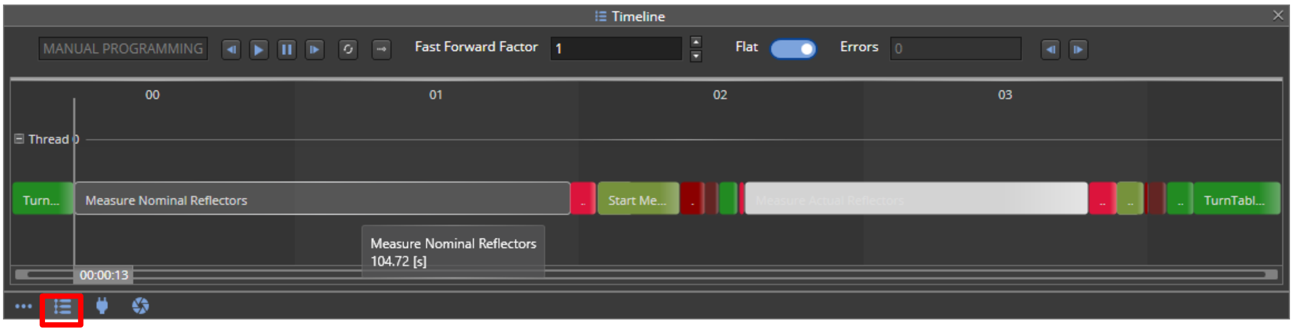


··· Status

Icon	Count	Time	Category	Message
ⓘ	1	01/17/2023 13:15:24	UI	Preparing cads for analysis
⚠	1	01/17/2023 13:17:06	UI	An attempt was made to prepare 'Leica_AT960_Head' for analysis, but it failed. The model is likely problematic and analysis may not work.
ⓘ	1	01/17/2023 13:17:06	UI	Finished preparing cads for analysis
⊗	1	01/17/2023 13:22:44	Collision	collided with , C0_XXXXXX_ML_CE001_IZQ_3, C0_XXXXXX_ML_RP021_00, C0_XXXXXX_ML_PL000_1RT_IZQ.stp.
ⓘ	1	01/17/2023 14:47:46	Execution	Execution started.
⊗	1	01/17/2023 14:47:50	Execution	Process was cancelled.
ⓘ	1	01/17/2023 14:47:50	Execution	Execution completed. Times: 00:00:07.80, 00:00:04.05
⊗	1	01/17/2023 14:47:50	Execution	Execution failed. : Process was cancelled.

4.8.2 Timeline

The timeline is a program's visual representation of the commands on an estimated event scale. The timeline is only accessible once the user enabled it (*RA* | *Settings* | *Docking* | *Timeline*) and after the program itself has run through validation.
Thanks to the timeline consisting of horizontal stacked bars, a user can skip commands or even fast forward to certain events.
Additionally, the timeline can show nested commands (pre-conditions) within other commands by disabling the **Flat** slider (e.g., a forced **Lock On** command during a **Start Measurement** command).
The user can also double-click on a timeline bar to highlight the corresponding command line in the related program.



Timeline

MANUAL PROGRAMMING

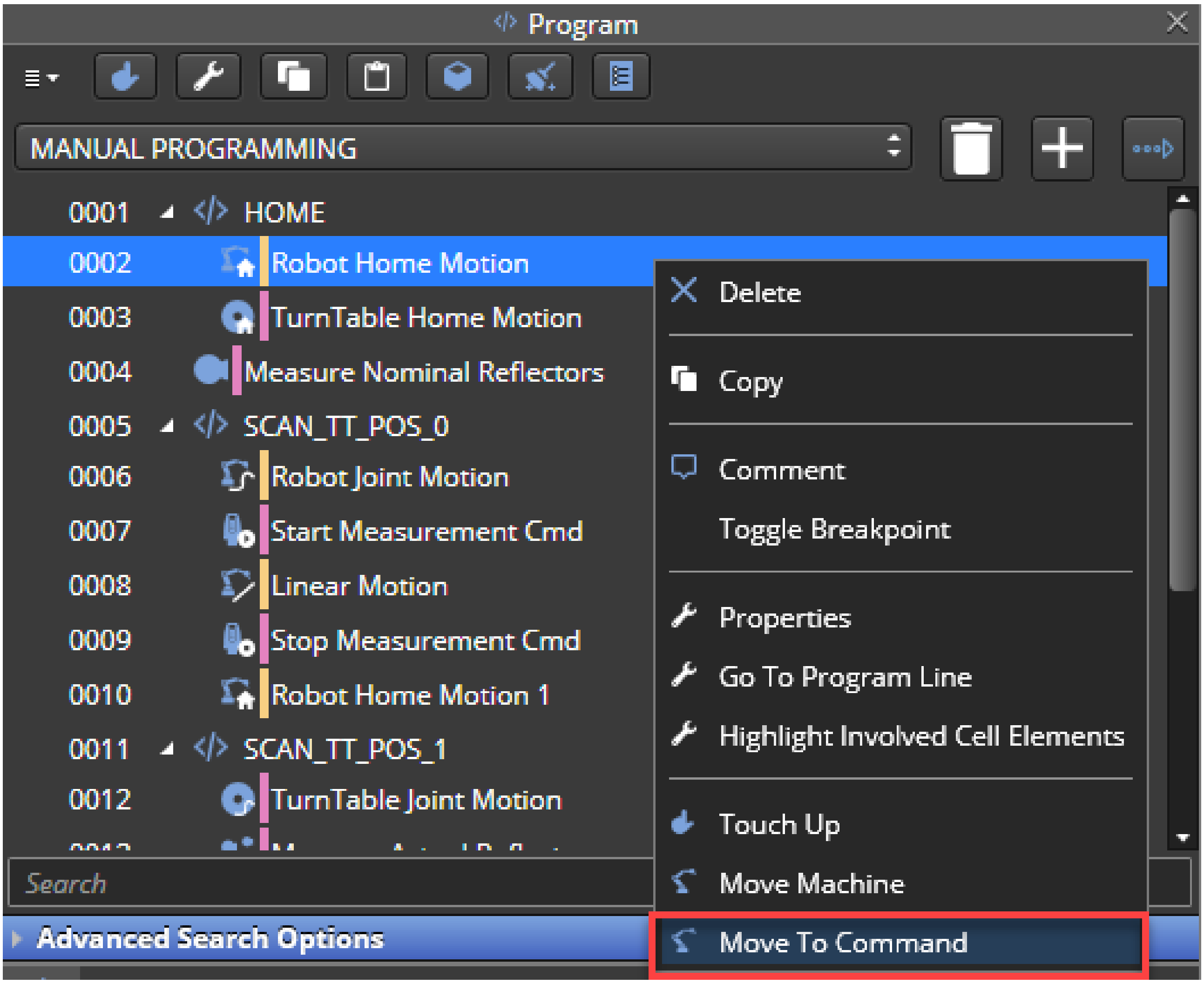
Fast Forward Factor 1

Flat

Errors 0

00
01
02
03
Thread 0
Turn...
Measure Nominal Reflectors
Start Me...
Measure Actual Reflectors
TurnTabl...
Measure Nominal Reflectors
104.72 [s]
00:00:13

Similarly, from the program view, it is possible to right-click on a command line and click on Move To Command to be pointed to the corresponding bar in the Timeline.

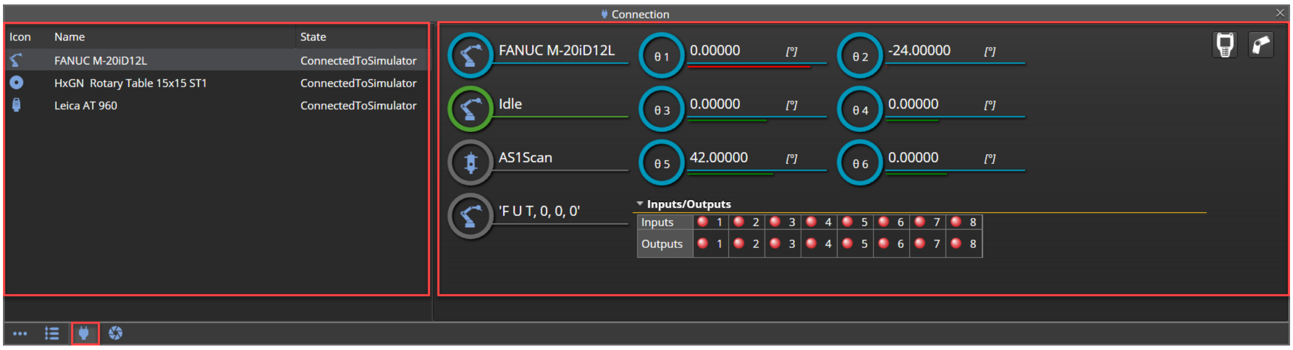


0001 HOME
0002 Robot Home Motion
0003 TurnTable Home Motion
0004 Measure Nominal Reflectors
0005 SCAN_TT_POS_0
0006 Robot Joint Motion
0007 Start Measurement Cmd
0008 Linear Motion
0009 Stop Measurement Cmd
0010 Robot Home Motion 1
0011 SCAN_TT_POS_1
0012 TurnTable Joint Motion

4.8.3 Connection

The Connection window will show:

1) on the left side, the various cell items including their icon, name, and connection state
2) on the right side, the selected item details. By default the item coordinates are shown in joint space (e.g. for a 6 DOF robot all the 6 joint values).



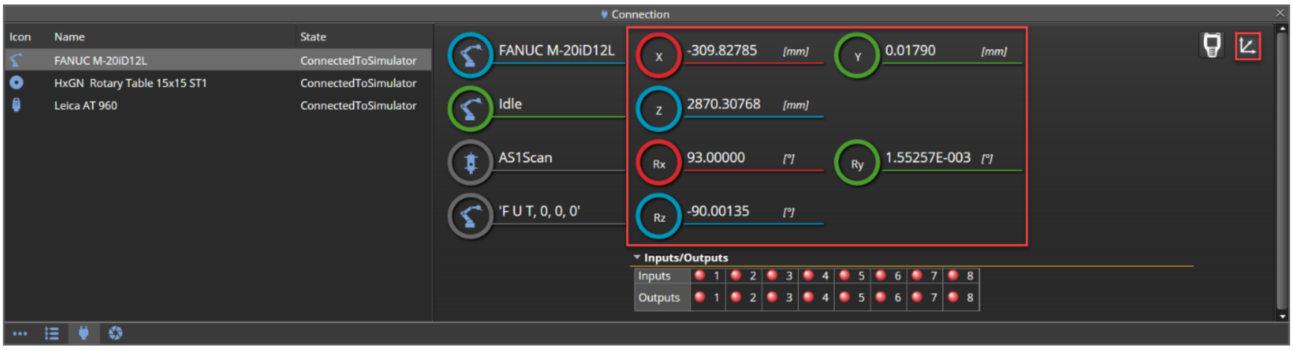
Icon	Name	State
	FANUC M-20iD12L	ConnectedToSimulator
	HxGN Rotary Table 15x15 ST1	ConnectedToSimulator
	Leica AT 960	ConnectedToSimulator

FANUC M-20iD12L
Idle
AS1Scan
'F U T, 0, 0, 0'

θ1 0.00000 [°]
θ2 -24.00000 [°]
θ3 0.00000 [°]
θ4 0.00000 [°]
θ5 42.00000 [°]
θ6 0.00000 [°]

Inputs	● 1	● 2	● 3	● 4	● 5	● 6	● 7	● 8
Outputs	● 1	● 2	● 3	● 4	● 5	● 6	● 7	● 8

To visualize the position of the robot TCP in World coordinates, it is possible to toggle the button. To restore the joint value view it is possible to toggle again the button.



Connection

Icon	Name	State
	FANUC M-20iD12L	ConnectedToSimulator
	HxGN Rotary Table 15x15 ST1	ConnectedToSimulator
	Leica AT 960	ConnectedToSimulator

FANUC M-20iD12L
Idle
AS1Scan
'F U T, 0, 0, 0'

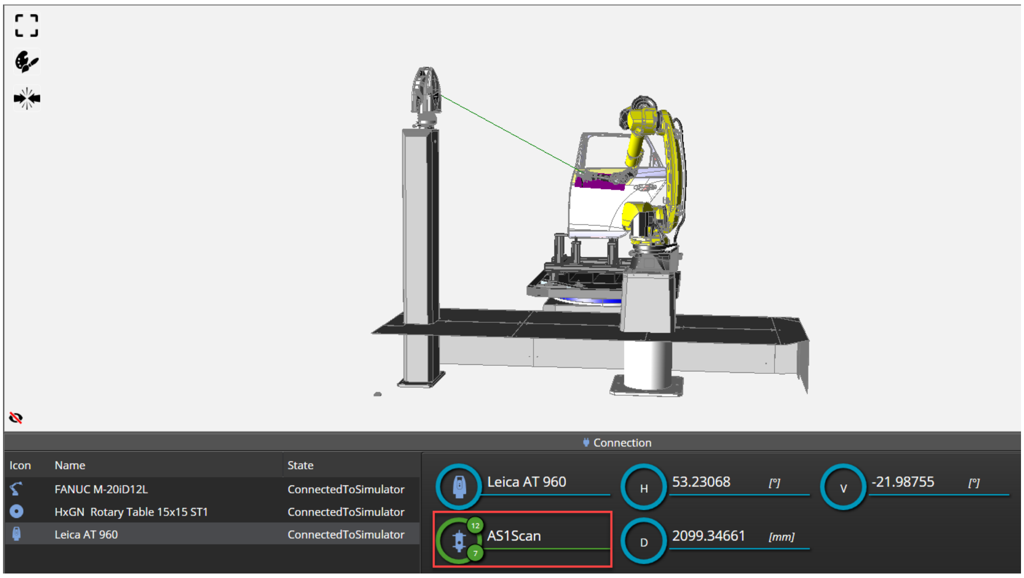
X -309.82785 [mm]
Y 0.01790 [mm]
Z 2870.30768 [mm]
Rx 93.00000 [°]
Ry 1.55257E-003 [°]
Rz -90.00135 [°]

Inputs/Outputs

Inputs	● 1	● 2	● 3	● 4	● 5	● 6	● 7	● 8
Outputs	● 1	● 2	● 3	● 4	● 5	● 6	● 7	● 8

The cell item details will also show more information depending on the item selected.

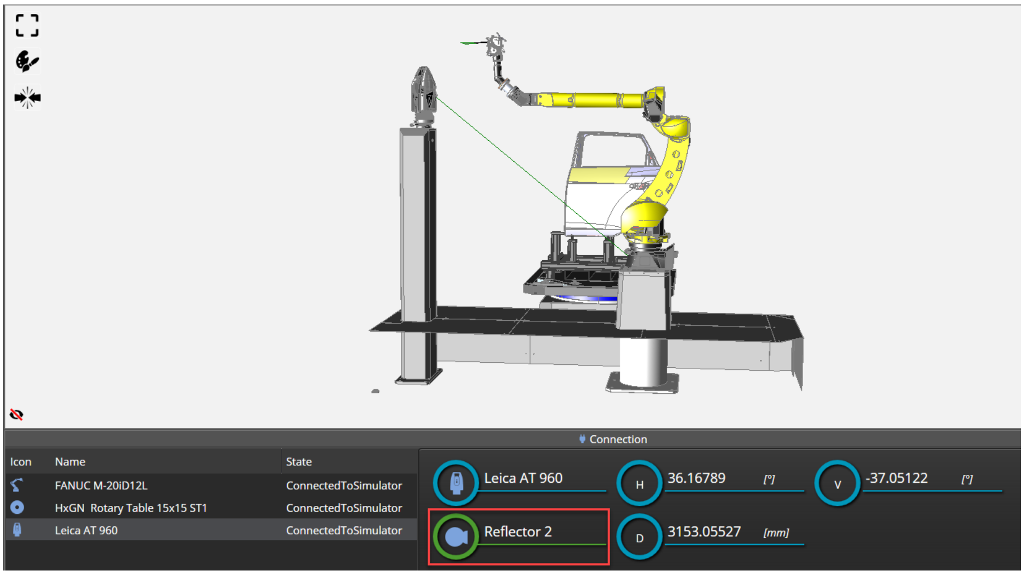
- The robot will show the tool connected and the inputs and outputs.
- The tracker will show the scanner face or the reflector to which is locked to.



Connection

Icon	Name	State
	FANUC M-20iD12L	ConnectedToSimulator
	HxGN Rotary Table 15x15 ST1	ConnectedToSimulator
	Leica AT 960	ConnectedToSimulator

Leica AT 960
AS1Scan
H 53.23068 [°]
V -21.98755 [°]
D 2099.34661 [mm]



Connection

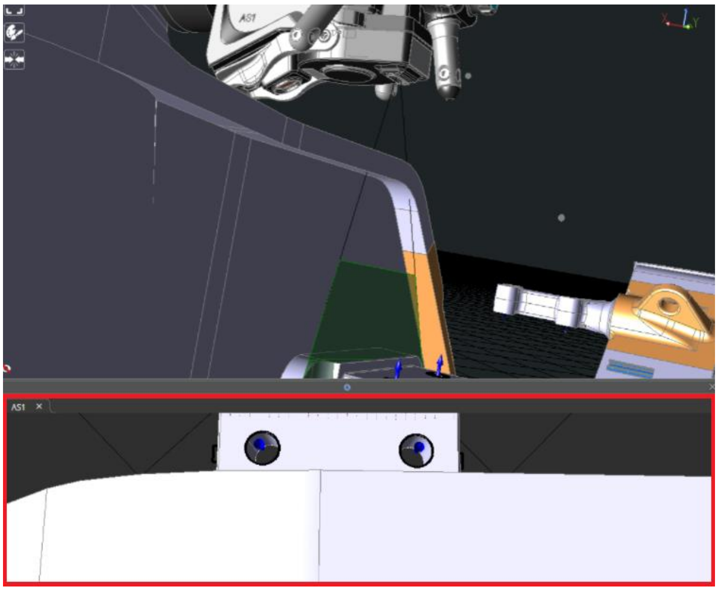
Icon	Name	State
	FANUC M-20iD12L	ConnectedToSimulator
	HxGN Rotary Table 15x15 ST1	ConnectedToSimulator
	Leica AT 960	ConnectedToSimulator

Leica AT 960
Reflector 2
H 36.16789 [°]
V -37.05122 [°]
D 3153.05527 [mm]

Additionally, the window will have an option to access the teach pendant for the selected cell item by clicking the button.

4.8.4 GoPro

The GoPro window is a viewport from either the tracker or the scanner's point of view. This window can be activated through the teach pendant.



4.9 Properties Window

This window will show the properties of the selected item in the cell or program command and in general is used to allow the user to specify the relevant parameters of the item or action that is being executed.

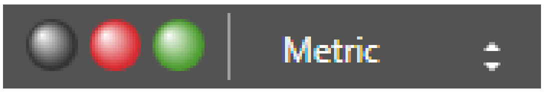
4.10 Metrology / Validation Engine / Collision Engine Buttons

This area has three buttons: starting from the left, the first button is used to connect with a **Metrology application**, the second one to interact with the **Validation engine** and the third one with the **Collision engine**.

The buttons have three possible colors:

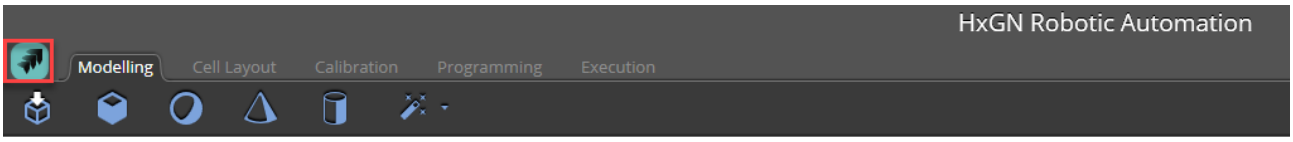
1) **Grey**: this means that HxGN Robotic Automation is not connected to the metrology SW or Validation/Collision engines are disabled
2) **Green**: this means that HxGN Robotic Automation is connected to the metrology SW or Validation/Collision engines are ready.
3) **Red**: this means an error happened connecting with the metrology SW or Validation/Collision engines are not ready.

Additionally, in the right part the user can define the measurement unit system between Metric, Imperial or Custom using a drop-down menu.



4.11 RA Main Menu Button

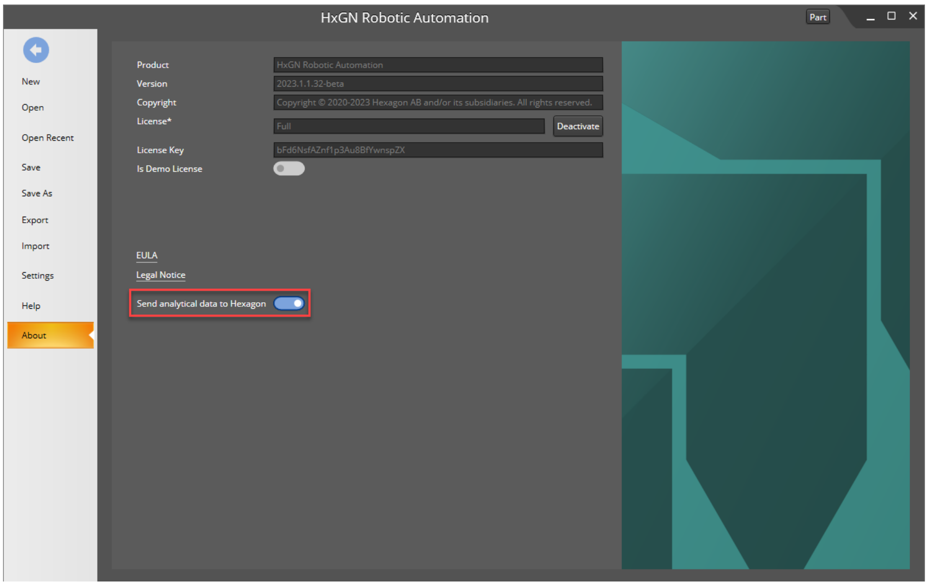
The RA main menu can be opened by clicking on the RA icon (▣) on the top-left corner of the screen.



The possible actions are shown in the picture below in the left menu available on light grey background and are self-explanatory.

1) **New**, to create a new project
2) **Open**, to open an existing project
3) **Open Recent**, to open an existing recent project
4) **Save**, to save the project
5) **Save As**, to save the project as a new project specifying a new name
6) **Export**, to save the project in a compressed .zip file format
7) **Import**, to extract a project from a compressed .zip file format previously exported.
8) **Settings**, to navigate to the HxGN Robotic Automation settings, described in detail in chapter 4.11.1.
8) **Help**, to be redirected to the web help.
9) **About** section has information about the HxGN Robotic Automation version installed and the license key in use.
   User can also consult the EULA and the Legal Notice about 3rd party software in use by clicking the EULA and Legal Notice buttons respectively.
   User can opt-in for sending analytical data to Hexagon to improve HxGN Robotic Automation by enabling the relative toggle switch. If enabled, the diagnostics collects crashes and errors, reporting the following data about the event occurred:
   - Device name
   - OS
   - User ID
   - Country
   - Language
   - Date and Time

   When any personal data is collected, such personal data will be processed in accordance with section 9.3 of EULA.



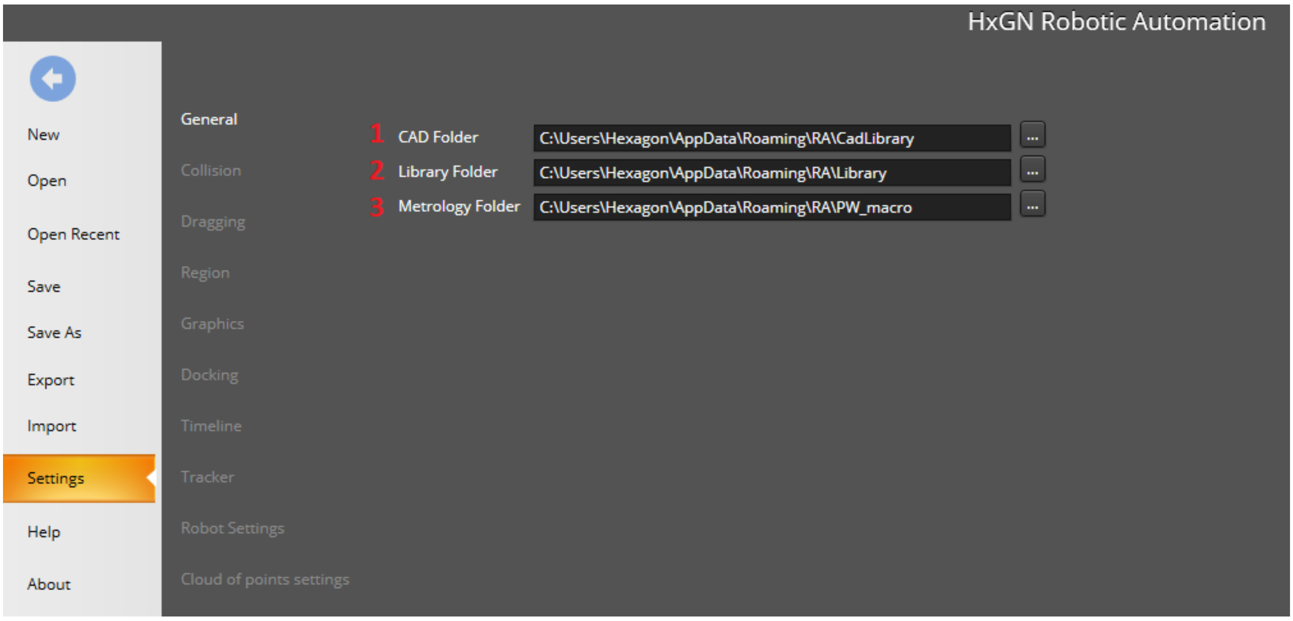
Product: HxGN Robotic Automation
Version: 2023.1.1.32-beta
Copyright: Copyright © 2020-2023 Hexagon AB and/or its subsidiaries. All rights reserved.
License*: Full
License Key: bFd6NsfAZnf1p3Au8BfYwnspZX
Is Demo License: ○
Send analytical data to Hexagon: ●

4.11.1 Settings

Within the RA main menu, it is possible to define the global settings of HxGN Robotic Automation in the different sections described below.

4.11.1.1 General

The general settings have folder locations where:
1) **CAD Folder**: store/load the CAD folders.
2) **Library Folder**: store/load the cell item library folders.
3) **Metrology Folder**: metrology-specific items location (e.g., PolyWorks Macros are stored here)



CAD Folder C:\Users\Hexagon\AppData\Roaming\RA\CadLibrary
Library Folder C:\Users\Hexagon\AppData\Roaming\RA\Library
Metrology Folder C:\Users\Hexagon\AppData\Roaming\RA\PW_macro

4.11.1.2 Collision

To enable collision detection using a switch button.

4.11.1.3 Dragging

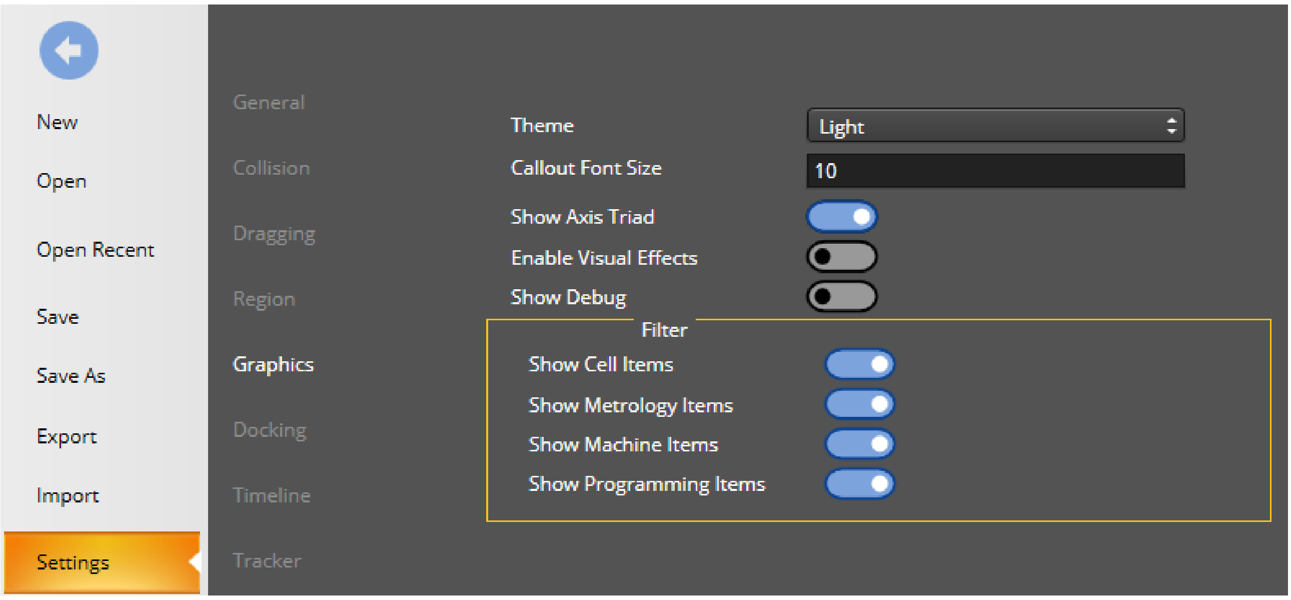
Advanced settings for the dragging mode in the 3D Window.
1) **Snapping Angle**: default value is 5°
2) **Snapping Length**: default value is 10 mm.
3) **Snapping Proximity**: default value is 1000 mm.

4.11.1.4 Region

To change the Language settings (e.g., English).

4.11.1.5 Graphics

To change advanced graphics settings as depicted in the picture below. The Theme change is currently not supported, only Dark theme is available. Enable Visual Effect will improve the visual representation of the cell items but may affect the performances.



Theme: Light

Callout Font Size: 10

Show Axis Triad: ●

Enable Visual Effects: ○

Show Debug: ○

Filter

Show Cell Items: ●

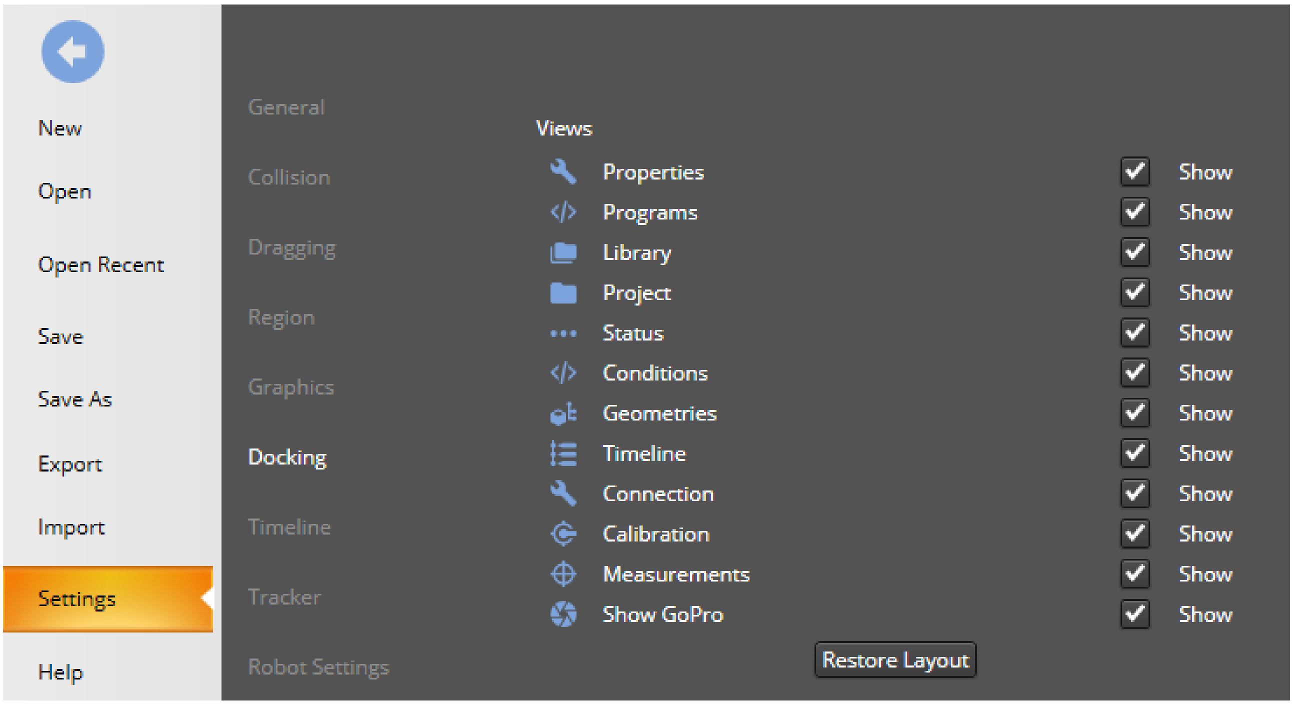
Show Metrology Items: ●

Show Machine Items: ●

Show Programming Items: ●

4.11.1.6 Docking

HxGN Robotic Automation application GUI (Graphical User Interface) has several panels showing in different Tabs (Modelling, Cell Layout, Calibration, Programming, or Execution). The user has the option to Show that panel or Restore Layout to default.

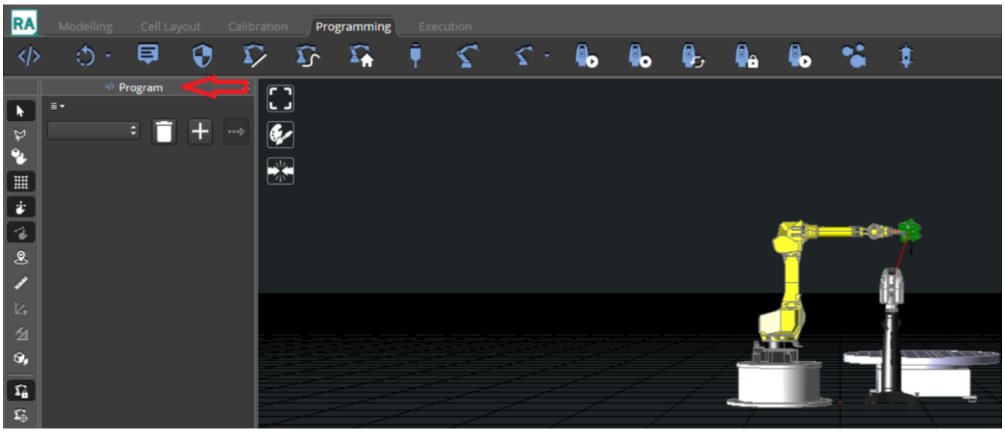


Views

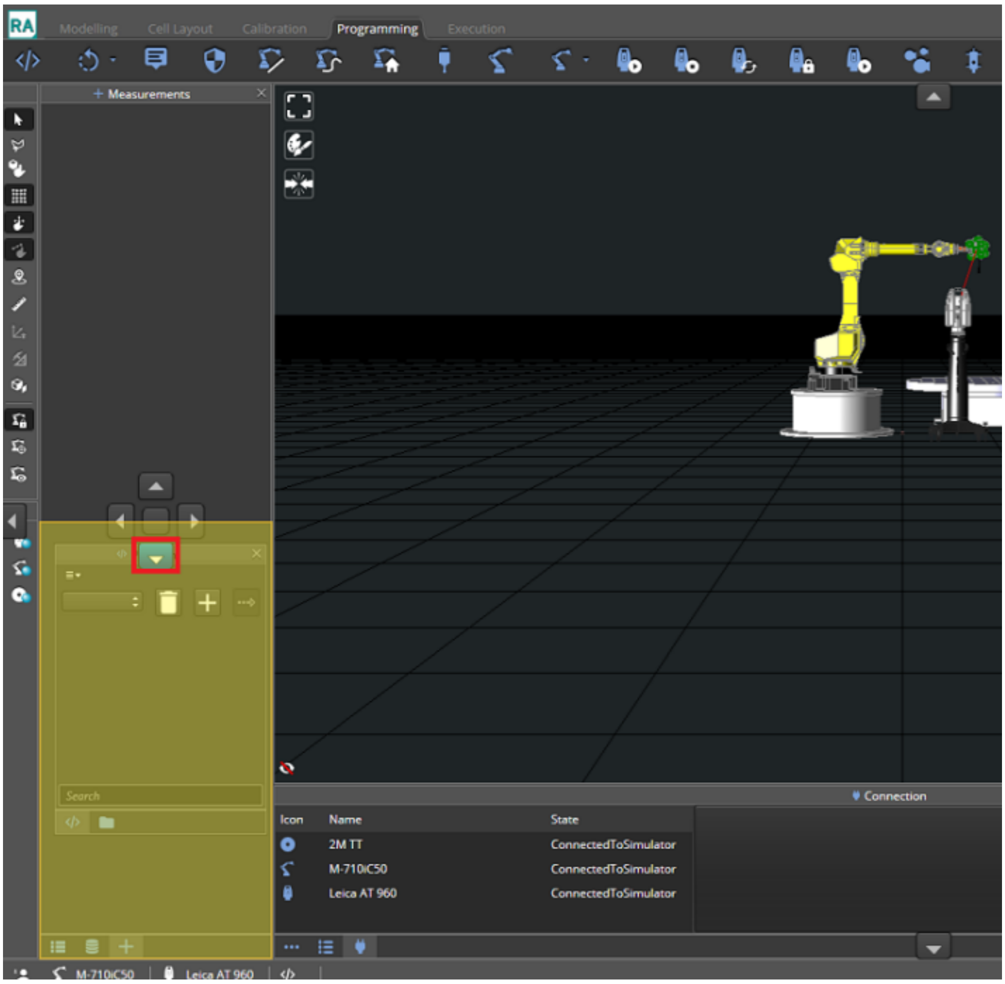
Properties	☑	Show
Programs	☑	Show
Library	☑	Show
Project	☑	Show
Status	☑	Show
Conditions	☑	Show
Geometries	☑	Show
Timeline	☑	Show
Connection	☑	Show
Calibration	☑	Show
Measurements	☑	Show
Show GoPro	☑	Show

To rearrange a panel, click on the panel with the mouse, hold the left mouse button and drag the panel to any location. Releasing the left mouse button will place it in the new desired location.

For example, if you want to change the programming panel, click where the red arrow with mouse. Hold the left button and drag it; a preview of new location will be shown in yellow, if you position it above one of the arrows:



Once you release the left mouse button, it will be positioned in the desired new location. In any stage, use can go to *Settings* | *Docking* and click on Restore Layouts.



Icon	Name	State
●	2M TT	ConnectedToSimulator
	M-710iC50	ConnectedToSimulator
	Leica AT 960	ConnectedToSimulator

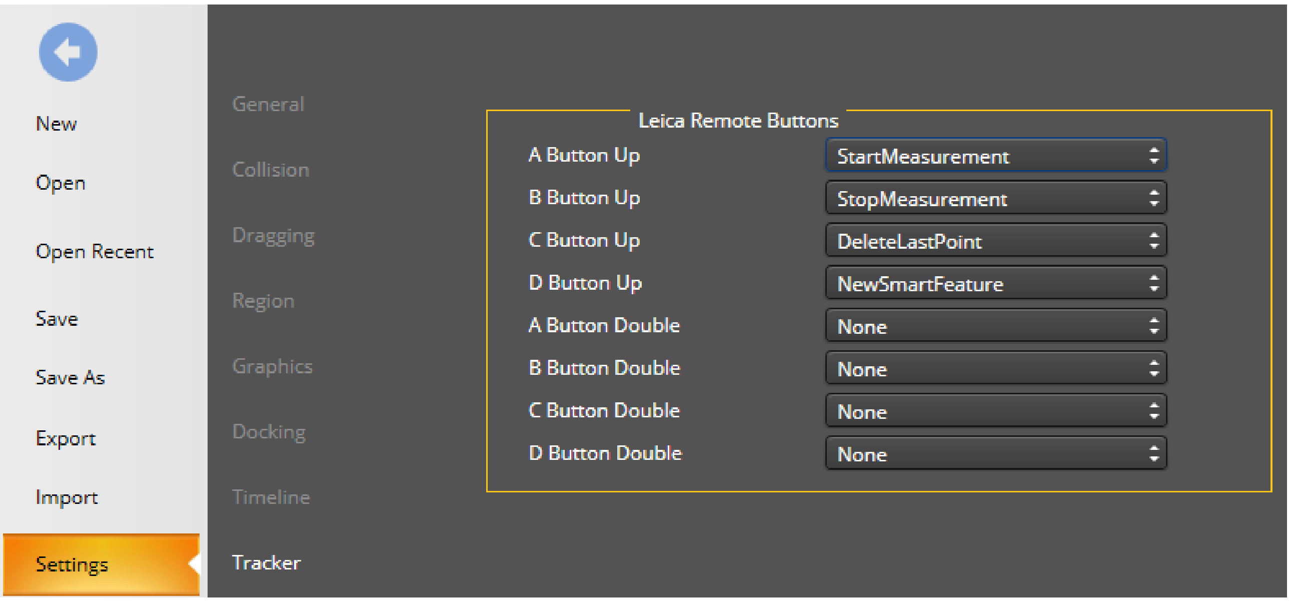
4.11.1.7 Timeline

Advanced settings for Timeline to specify:

1) Min Time bar Selection Range: default value 0.1 s
2) Min Time bar Selection Range: default value 5 s

4.11.1.8 Tracker

This section is used to configure Leica Remote Buttons when using a Leica sensor, e.g. using a T-Probe for starting a measurement by clicking the A Button.



Leica Remote Buttons

A Button Up	StartMeasurement
B Button Up	StopMeasurement
C Button Up	DeleteLastPoint
D Button Up	NewSmartFeature
A Button Double	None
B Button Double	None
C Button Double	None
D Button Double	None

4.11.1.9 Robot settings

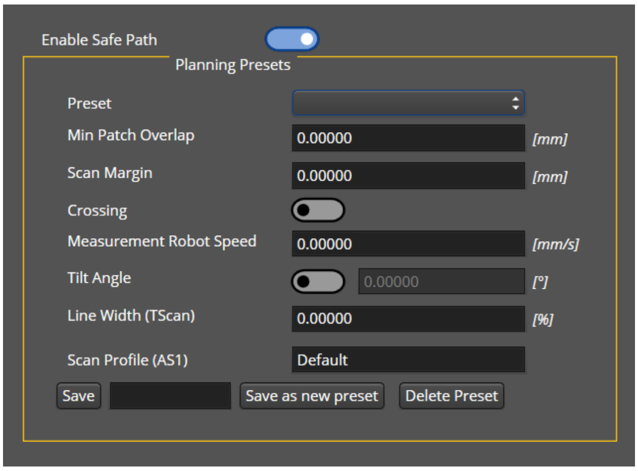
This section allows to modify default Joint Motion Speed (10% override) and Linear Motion Speed (400 mm/s).

4.11.1.10 Cloud of points settings

This section allows to change the COP sparsity (e.g. 1 means that 100% of points will be shown, 10 means 10% of points will be shown) and visualize or hide the point normal vectors.

4.11.1.11 Planning

In this area it is possible to define presets for the path planning algorithm to be used to scan optimally the supported metrology features (e.g. circle or slot). It is also possible to define a new custom preset.



Enable Safe Path

Planning Presets

Preset

Min Patch Overlap 0.00000 [mm]

Scan Margin 0.00000 [mm]

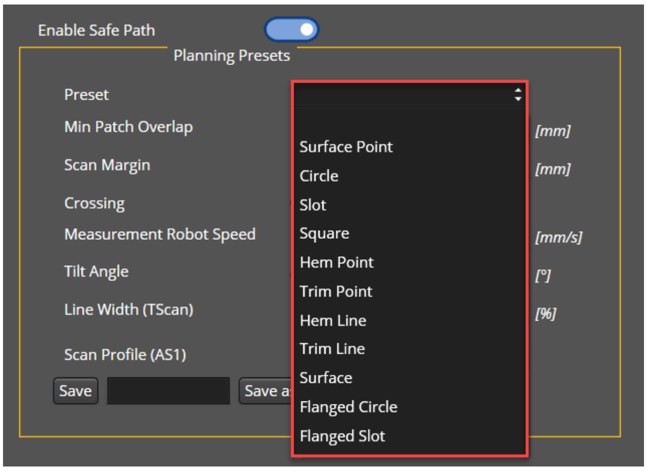
Crossing

Measurement Robot Speed 0.00000 [mm/s]

Tilt Angle 0.00000 [°]

Line Width (TScan) 0.00000 [%]

Scan Profile (AS1) Default



4.11.1.12 Colors

Define the colors used for the various elements of the application.

4.11.1.13 Sounds

Enable Sounds to emit the same sound emitted by the real AT960 sensor when a measurement is collected.

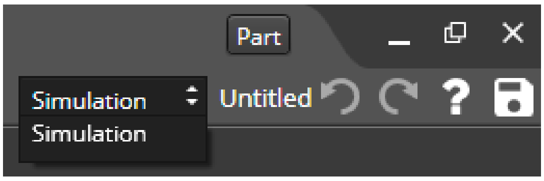
4.11.1.14 Import

Auto rename option in case the imported project has the same name, following Windows convention.

4.12 Top-right Menu

The top right area allows to connect with simulation or real cell installations using the drop-down menu, undo and redo changes made on the project currently in use, save it with the relative icon and perform the usual operations as minimize, expand and close the HxGN Robotic Automation application.

The Part button is related to Inspire metrology software interface and it is explained in chapter 10.2.1



4.13 Keyboard shortcuts

Action	Key Shortcut	Comment
Copy	Ctrl + C	
Paste	Ctrl + V	

Add Program	Ctrl + P	
Add Linear Motion	Ctrl + L	
Add Joint Motion	Ctrl + J	
Zoom Fit	Ctrl + F	On selected item. If no item is selected, general zoom.
Save	Ctrl + S	
Save As	Ctrl + Shift + S	
Open Project File	Ctrl + O	
New Project File	Ctrl + N	
Undo	Ctrl + Z	
Redo	Ctrl + Y	
Cancel Action	Esc	
Confirm Selection	Enter	
Remove Selected Item	Del	
Rename	F2	
Reset Measurement	F3	
Start Measurement	F4	
Execute	F5	
Associate Measurement	F6	
Execute Next Step	F10	
Select Group	G	Hold G while on 3D window and it will select the group, not the item.
Execute Command	Tab + double-click	While on programing or execution modes
Clone Item	Alt + drag	
Expand Tree	Ctrl + “+”	
Collapse Tree	Ctrl + “-”	
Pick On CAD Rotate	Shift + drag	When on pick on cad mode is selected
Pick On CAD Flip Normal	Alt + drag	When on pick on cad mode is selected
Range Selection	Shift + click	
Multiple Selection	Ctrl + click	
Drag/Rotate by a Fixed Step	Shift + drag	Rotates at 5° for rotation and 10 mm for translations

5 Modelling

This section is aimed for **Application Engineers**. Users do not have to model robotic cells; they usually start using HxGN Robotic Automation already configured to work on the robotic cell installation where it is installed.

5.1 Machine creation

HxGN Robotic Automation application has the capability to model machines which include the following:

1) **Slider**: machine extension, which includes a fixed part and linear moving part to supply more DOFs (degree of Freedom) either to a robot or a tracker.

2) **Robot**: industrial anthropomorphic robots:
   a) Six axes robot (e.g., Kuka)
   b) Six axes coupled robot – where there is a dependency between joint J2 to joint J3. (e.g., Fanuc)

3) **Turn Table**: part holder, which includes a fixed part and a rotating moving part.

Prior to building the model the following must be prepared:

1) **CAD file** (format such as STP, JT, IGS, etc) that must be capable of being separated into groups for each joint.

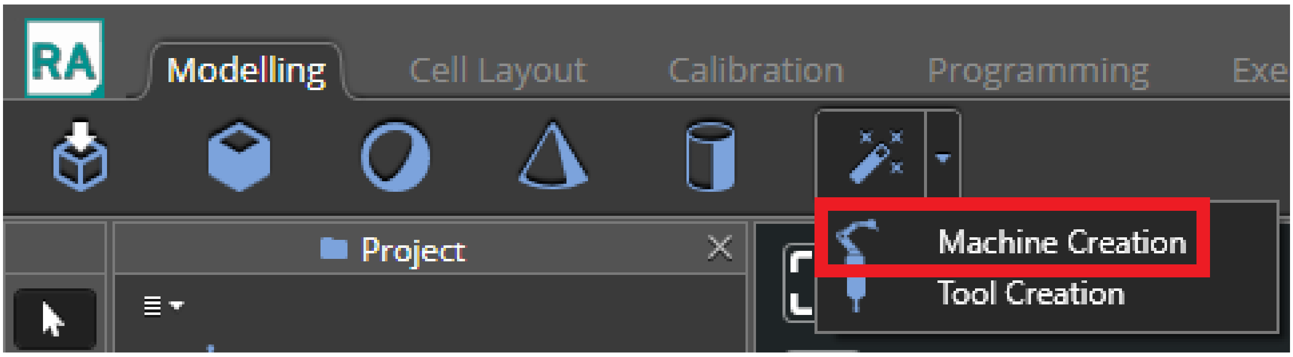
2) **Model name**, data sheet specification of the Robot or Part holder or Robot extension. This is usually a PDF file which has the specification drawing and dimensions, joint ranges, and velocities.

   When defining joints range parameters such as Min and Max, pay attention that it is not allowed to have joints limitations which are wider/higher than the limitation of the real robot.

3) **DH parameters** (for further explanation see chapter 5.1.9).

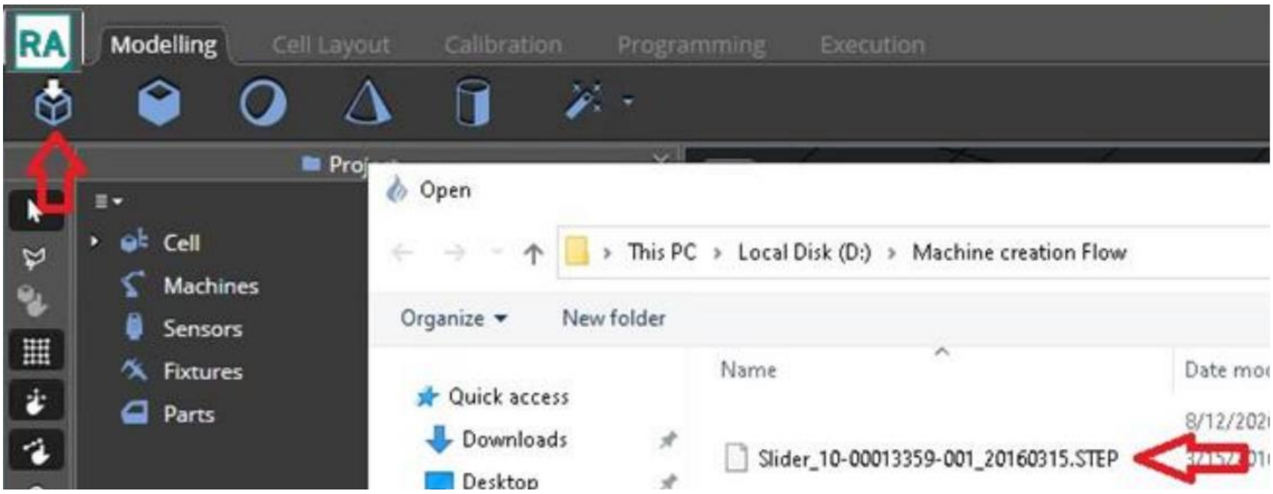
4) **Offset Transform parameters** (usually needed for slider).

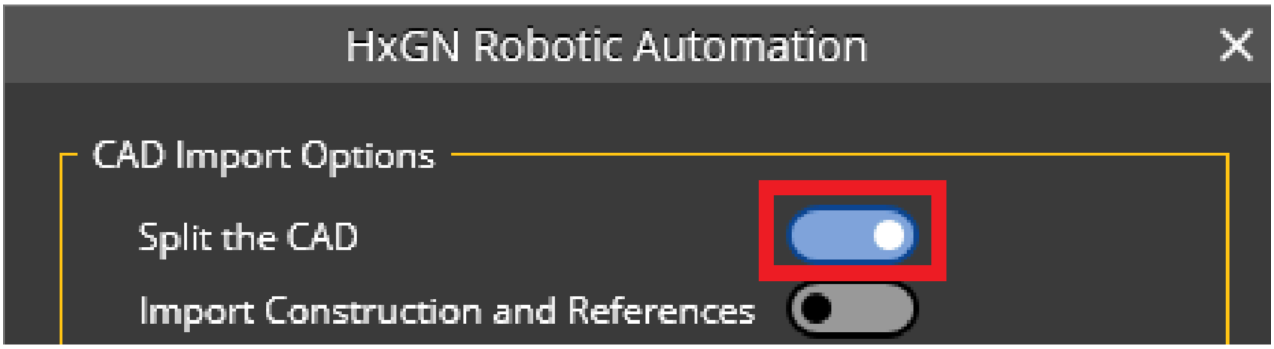
HxGN Robotic Automation also supplies a **Machine Creation** wizard to simplify the creation of machines.



5.1.1 Slider

Open a new project with Robotic Automation application and under **Modelling** tab, click on the **Import CAD** icon to add the slider's CAD file and check **Split the CAD** switch (disabled by default).



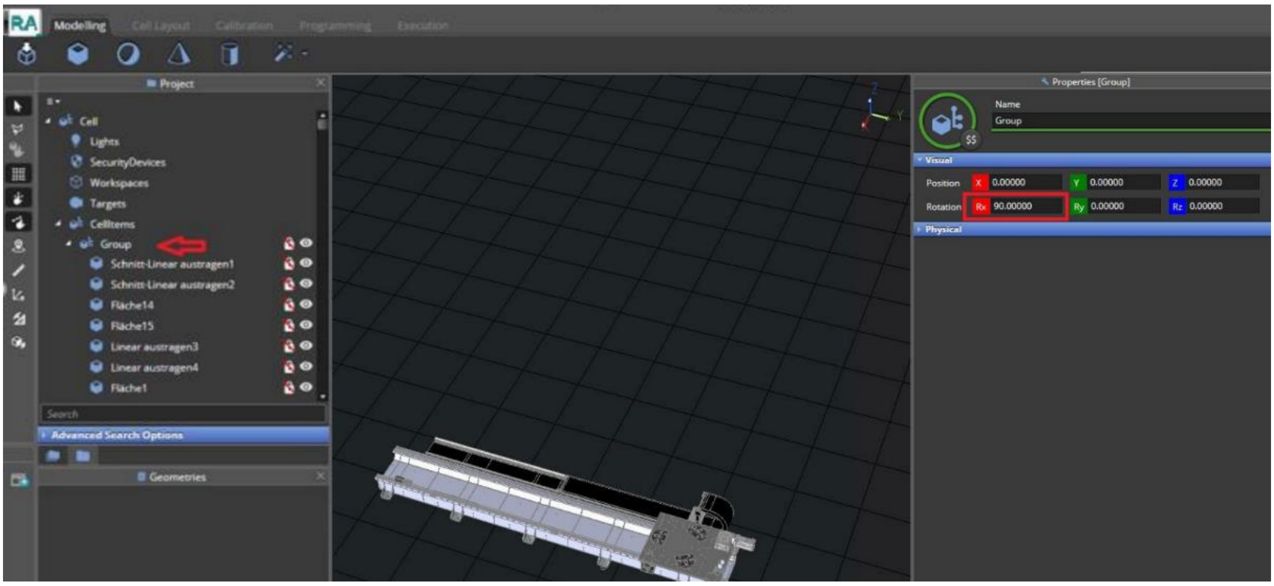

HxGN Robotic Automation

CAD Import Options

Split the CAD ●

Import Construction and References ○

The CAD will show under **Cell items** as a **Group**. The CAD coordinate system may need to be modified. Double click on the **Group**, the properties box will open on the right, edit the **R**$_X$, **R**$_Y$ or **R**$_Z$ fields, and make sure the slider moving part is up to Z direction (see the example below, **R**$_X$ should be 90°).

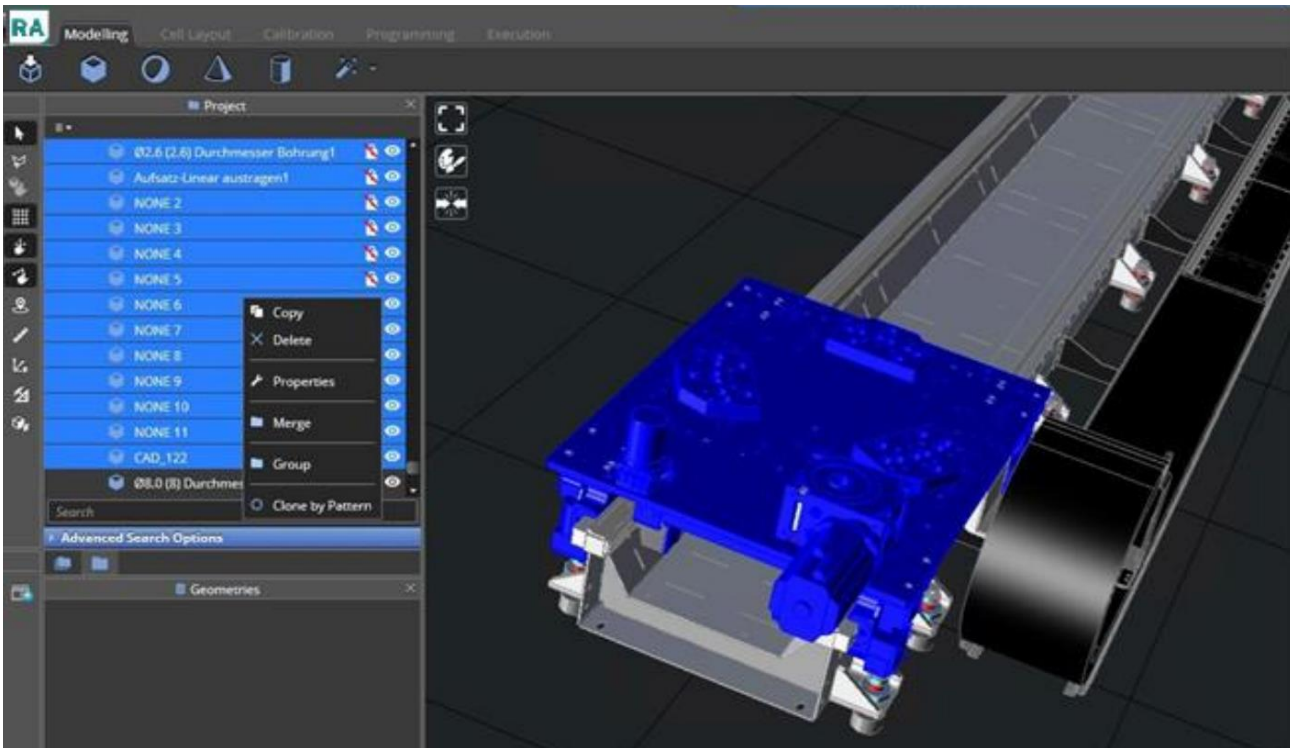


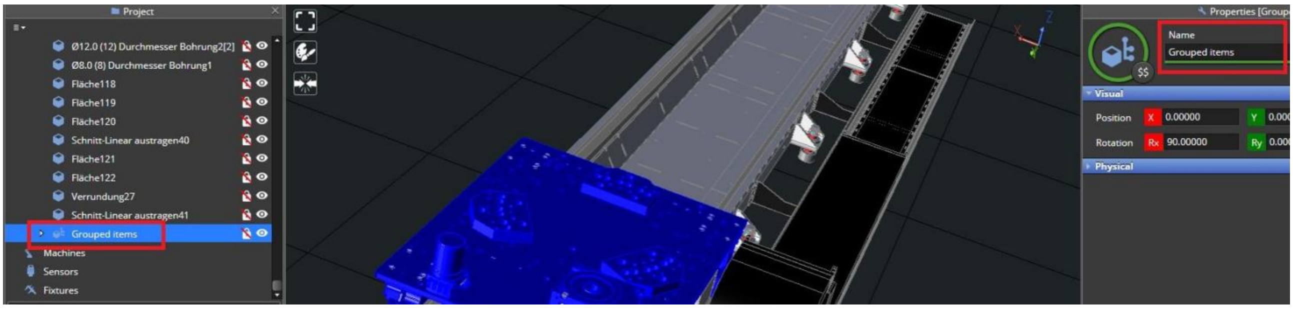
Name: Group

Position X 0.00000 Y 0.00000 Z 0.00000

Rotation Rx 90.00000 Ry 0.00000 Rz 0.00000

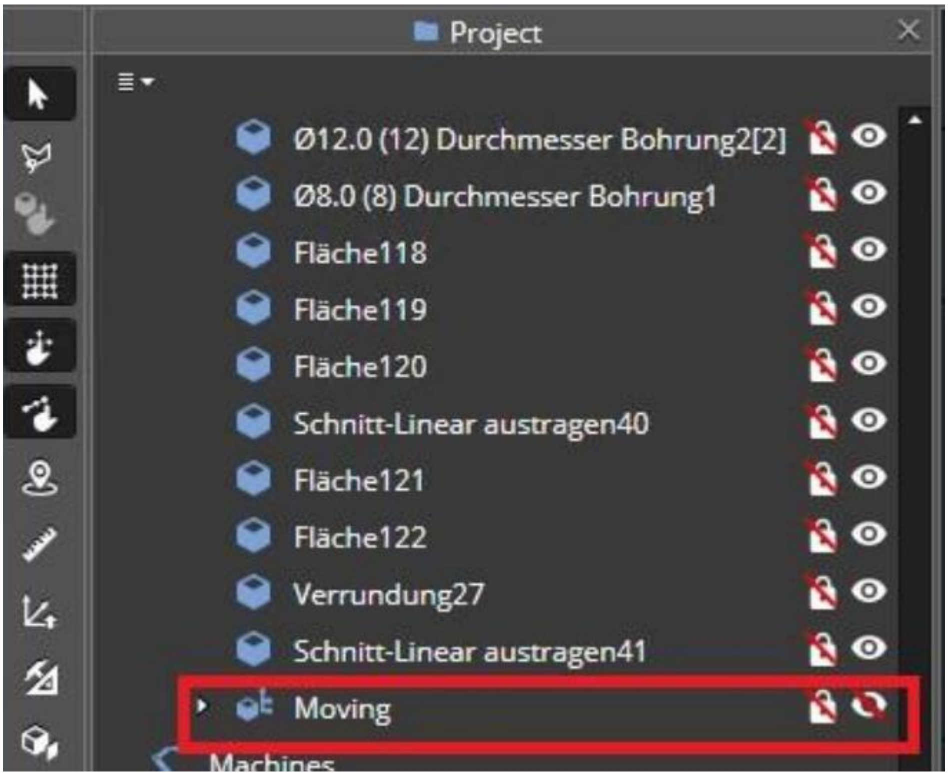
Within the group it is possible to remove unnecessary CADs and divide into two groups: fixed and moving.
After grouping correctly, the CADs in each group must be merged by right clicking on the **Group** and selecting **Merge**.
User can find the CAD elements, and by selecting the [CTRL] Key, right click and **Group**. Grouped CADs will be moved into **Grouped Items** in the project tree and can be renamed as “Moving” by double click and rename them in the Property window.





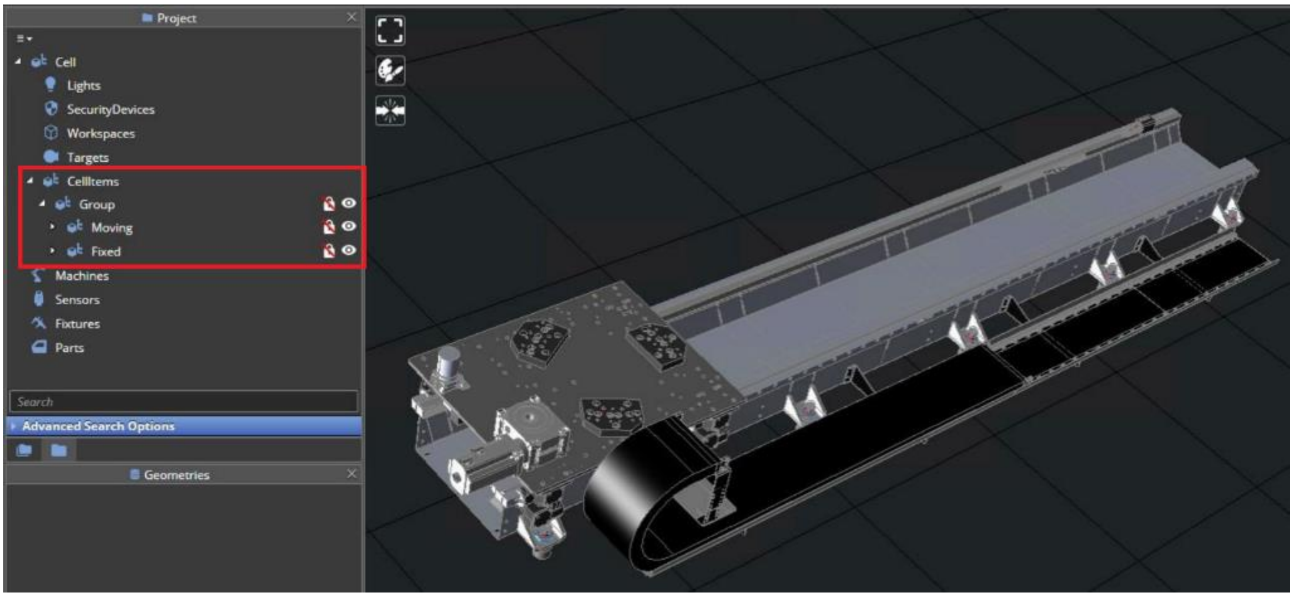
Name: Grouped items
Position X 0.00000 Y 0.000
Rotation Rx 90.00000 Ry 0.000

To group the fixed parts, user must hide the new “Moving” group that was just created.


Project
Ø12.0 (12) Durchmesser Bohrung2[2]
Ø8.0 (8) Durchmesser Bohrung1
Fläche118
Fläche119
Fläche120
Schnitt-Linear austragen40
Fläche121
Fläche122
Verrundung27
Schnitt-Linear austragen41
Moving
Machines

After removing all remaining unnecessary CADs, the user must select the remaining CAD items, group them, and rename the new group to “Fixed” in properties as done before.

The Moving group can now be shown again, and the two groups will be visible within the project tree.



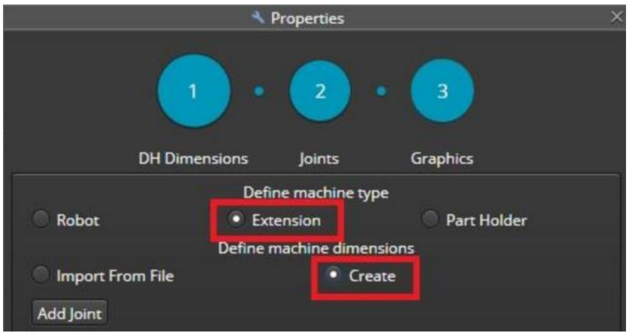
To complete the modelling, the two groups can be merged by right-clicking and selecting **Merge** and renamed again as “Moving” and “Fixed” as done in the earlier step.

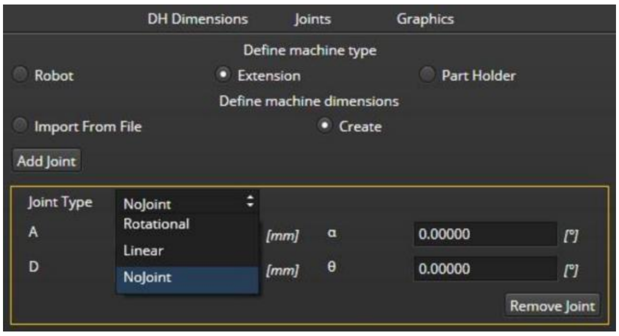
If CAD orientation has changed, it can be fixed by changing **R**$_X$ or **R**$_Y$ in the **Properties** window.

5.1.1.1 Slider creation wizard

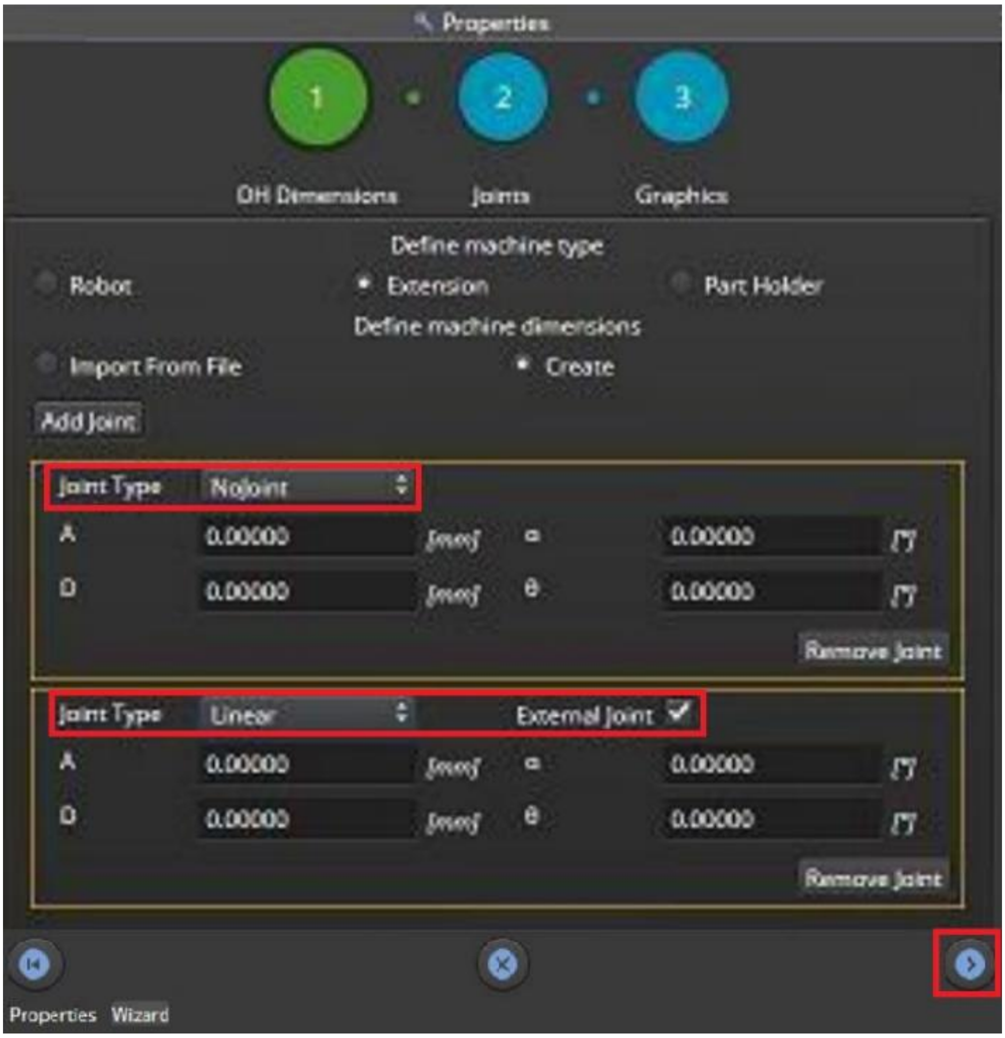
After clicking on the **Machine Creation** wizard, on the Properties window:

1) select **Extension** as machine type, select **Create** as machine dimensions, then click **Add Joint** and select **No Joint** from the drop-down menu.

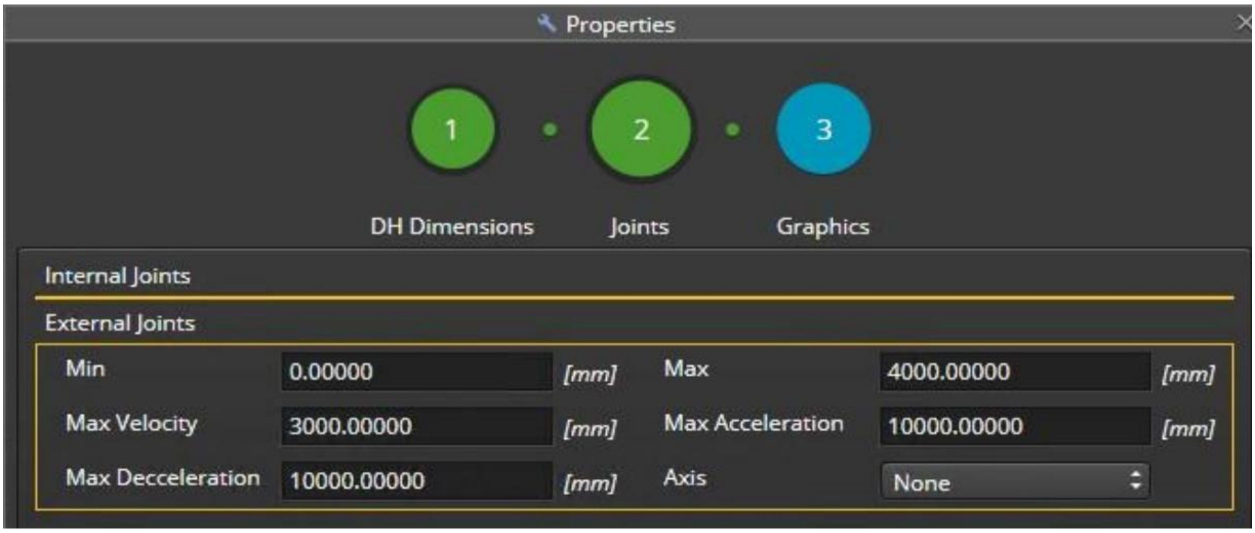




2) Click again on **Add Joint** to add another joint selecting **Linear** from the drop-down menu for the moving joint and checking **External Joint**. Once done, click on the arrow on the bottom right to continue to the next step.

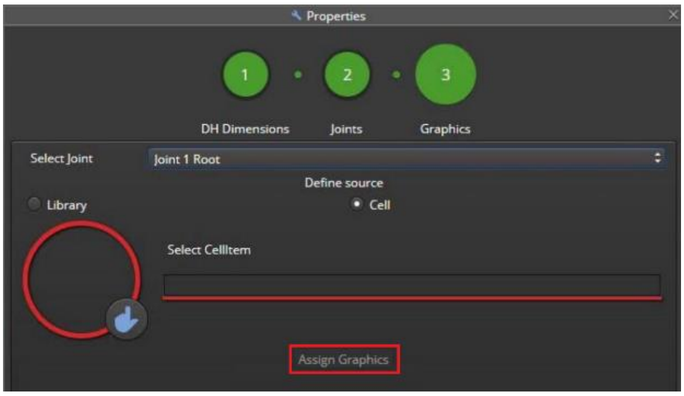


3) Define joints parameters such as Min and Max velocity. This step can be skipped and change later but always make sure that it is not allowed to have joints limitation which are outside of the real machine joint limits. Click again on the bottom-right corner to continue with the next step.



Min	0.00000	[mm]	Max	4000.00000	[mm]
Max Velocity	3000.00000	[mm]	Max Acceleration	10000.00000	[mm]
Max Decceleration	10000.00000	[mm]	Axis	None	

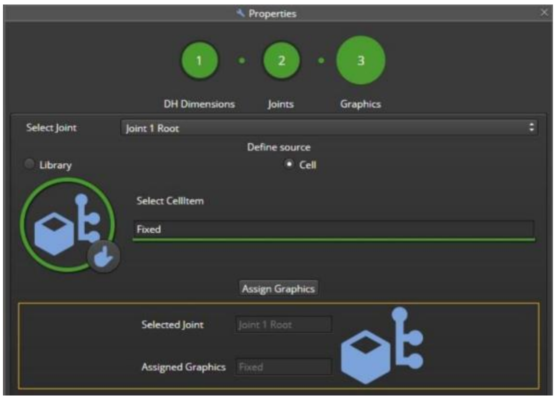
4) Assign graphics: from the **Select Joint** drop-down menu select **Joint 1 Root** first and then set **Define source** to **Cell**. Click on the hand icon to select the graphics and select the **Fixed** group under **Cell Items** group. It will turn its colour to green if the operation is successful and then by hitting the [ENTER] key the assignment will be confirmed. Click on **Assign Graphics** once you see the item was selected and you will see the assigned items in the list.



Select Joint: Joint 1 Root

Define source: ○ Library ● Cell

Select CellItem



Select Joint: Joint 1 Root

Define source: ○ Library ● Cell

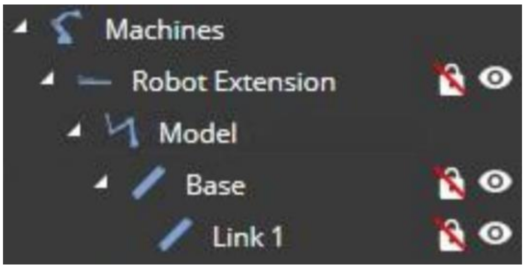
Select CellItem: Fixed

Selected Joint: Joint 1 Root

Assigned Graphics: Fixed

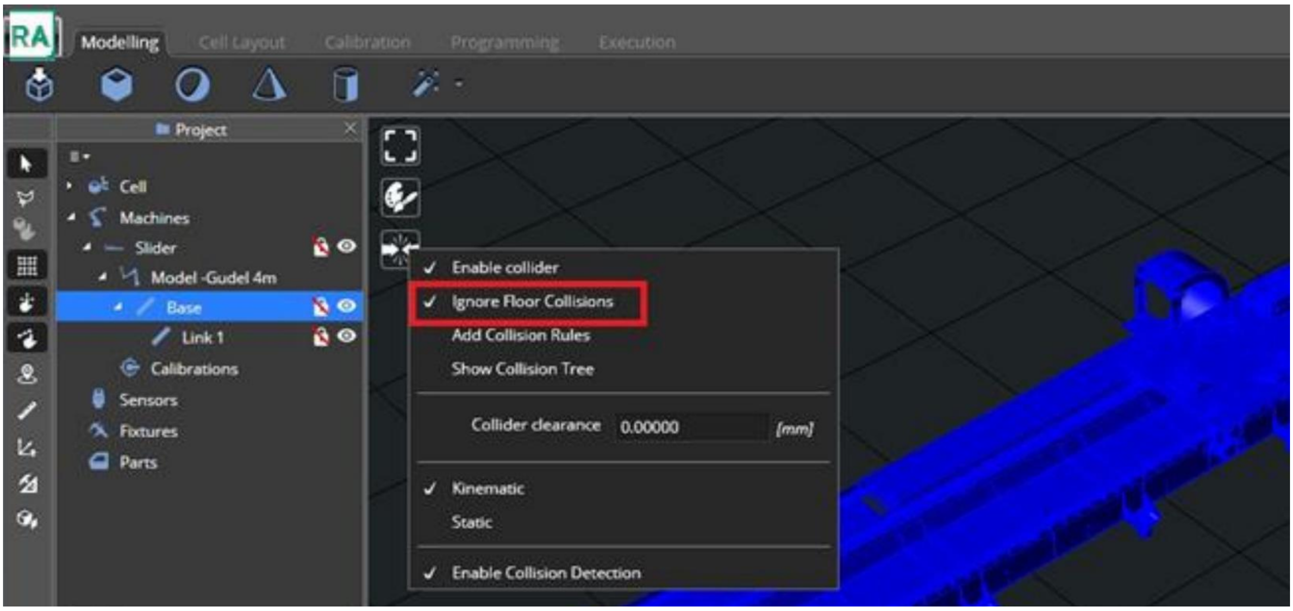
5) You can continue with the same steps for the other joint (**Joint 2 External**) by selecting it from the **Select Joint** drop-down menu and clicking on the **Finish** icon ( ) to complete the process.

At the end of the process under **Machines** in the project tree you will find the new machine. You can rename the Robot extension and Model names as you prefer.



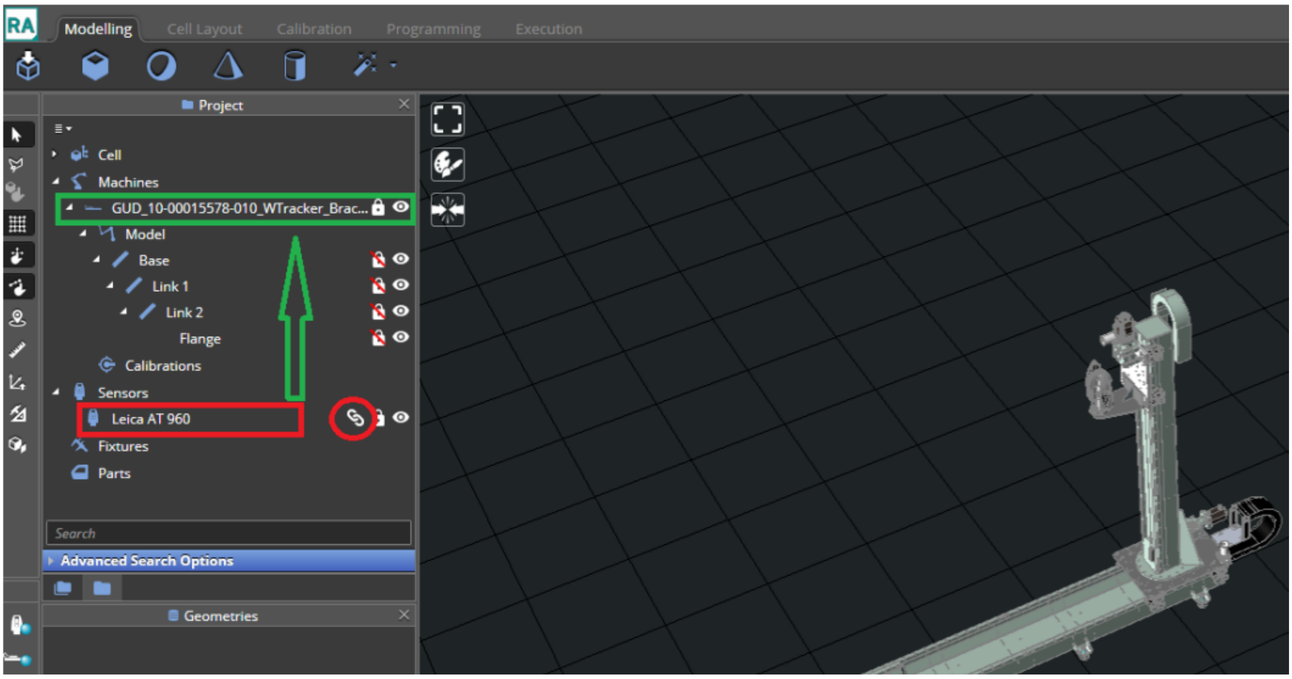
- Machines
  - Robot Extension
    - Model
      - Base
        - Link 1

In case you have a collision with the floor, you can select the slider **Base**, click on **Collision** icon and check **Ignore Floor Collisions**.

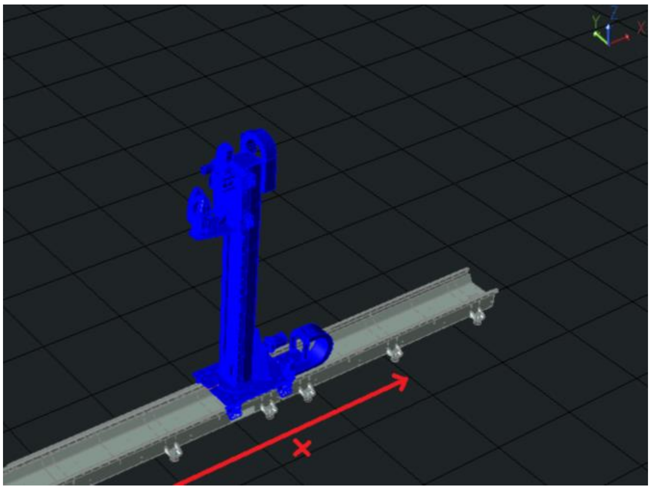


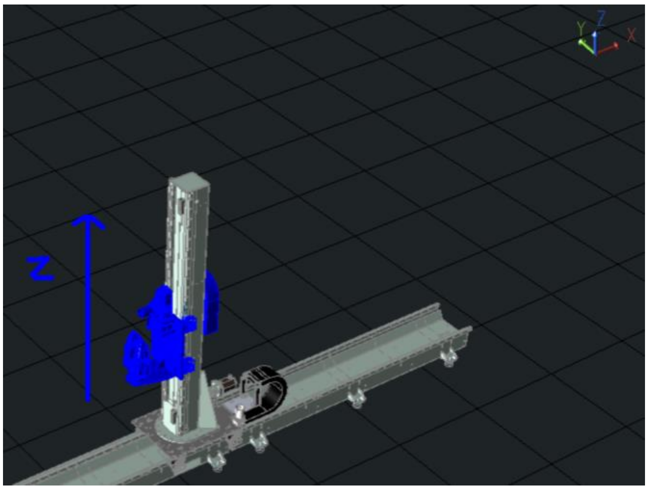
5.1.2 Tracker on slider

It is possible to model a slider (either with one or two axes) and link the Tracker to that slider by dragging and dropping it to the slider as depicted below.



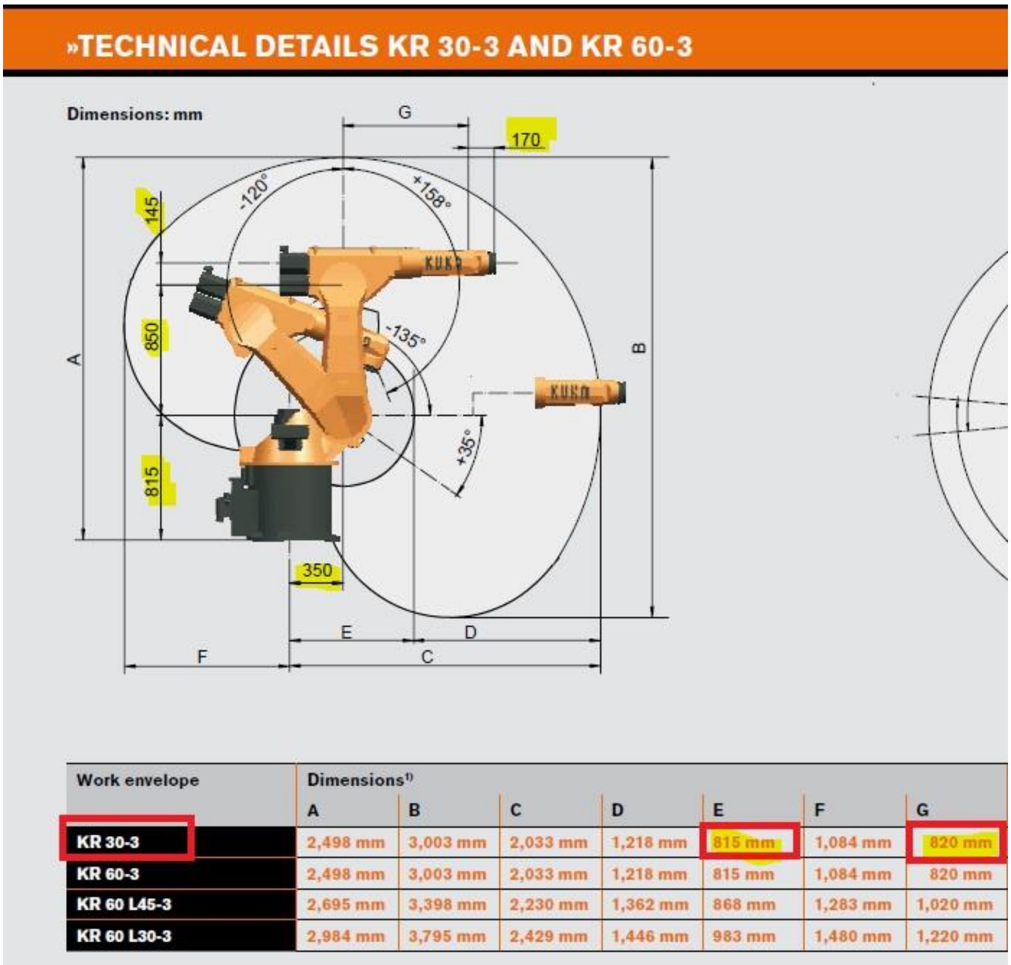
In this way the tracker can move together with the slider in X and Z directions.





5.1.3 Robot

Prior to Robot creation, it is needed to prepare the CAD and the specification data sheet to extract the D-H parameters. In the following, an example using a KUKA KR 30-3 robot will be used to show how to extract the D-H parameters from the robot data sheet.



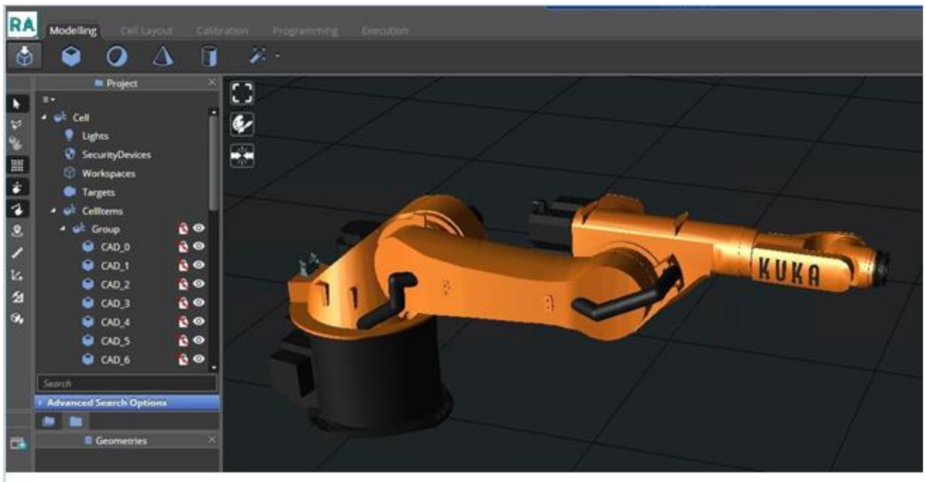
»TECHNICAL DETAILS KR 30-3 AND KR 60-3

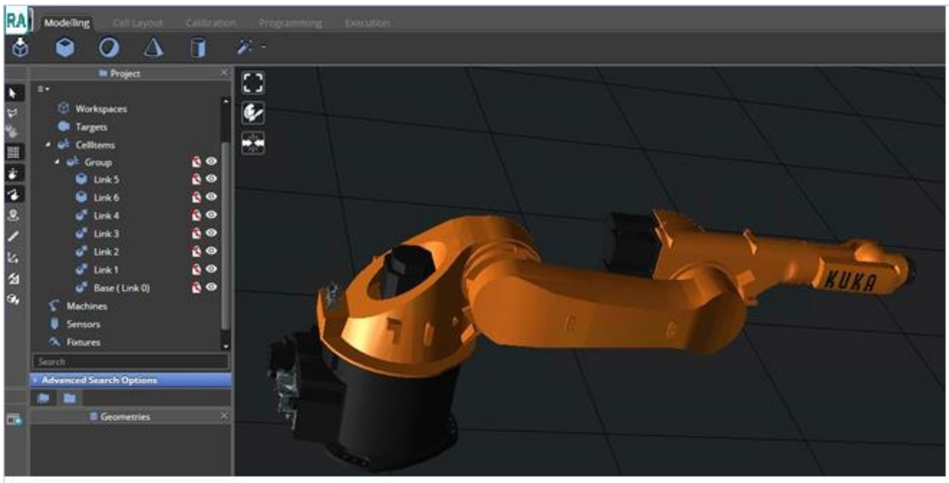
Work envelope	Dimensions[1]						
	A	B	C	D	E	F	G
KR 30-3	2,498 mm	3,003 mm	2,033 mm	1,218 mm	815 mm	1,084 mm	820 mm
KR 60-3	2,498 mm	3,003 mm	2,033 mm	1,218 mm	815 mm	1,084 mm	820 mm
KR 60 L45-3	2,695 mm	3,398 mm	2,230 mm	1,362 mm	868 mm	1,283 mm	1,020 mm
KR 60 L30-3	2,984 mm	3,795 mm	2,429 mm	1,446 mm	983 mm	1,480 mm	1,220 mm

Joint / Link	R (A)	D	α	θ (zero offset)
Base – Link 0	0	-815	180	--
J1 – Link 1	350	0	90	0
J2 – Link 2	850	0	0	0
J3 – Link 3	145	-820	90	-90
J4 – Link 4	0	0	-90	0
J5 – Link 5	0	-170	90	0
J6 – Link 6	Transform offset	--	----	----

Open a new project with Robotic Automation application and under **Modelling** tab, click on the import icon to add the robot’s CAD files.

Delete unnecessary CAD, separate, and classify them into merged cad group names as follows: Base (Link 0), Link 1, Link 2, Link 3, Link 4, Link 5, Link 6.

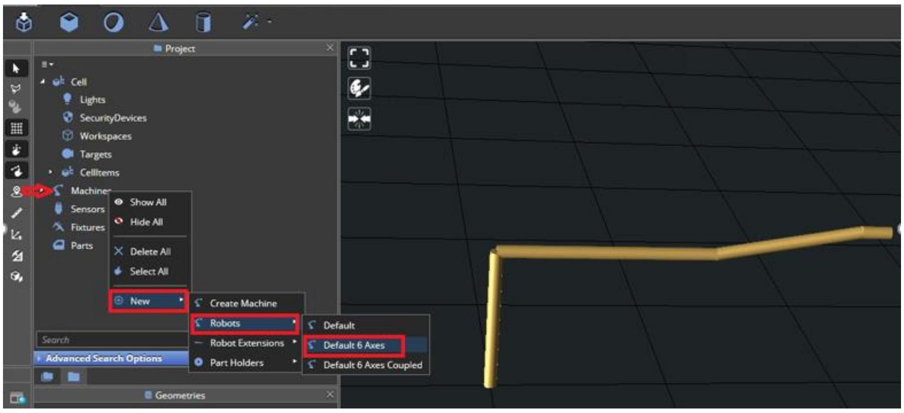


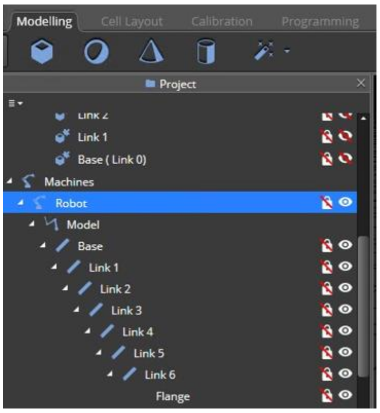


A robot can be created manually or following the robot creation wizard.

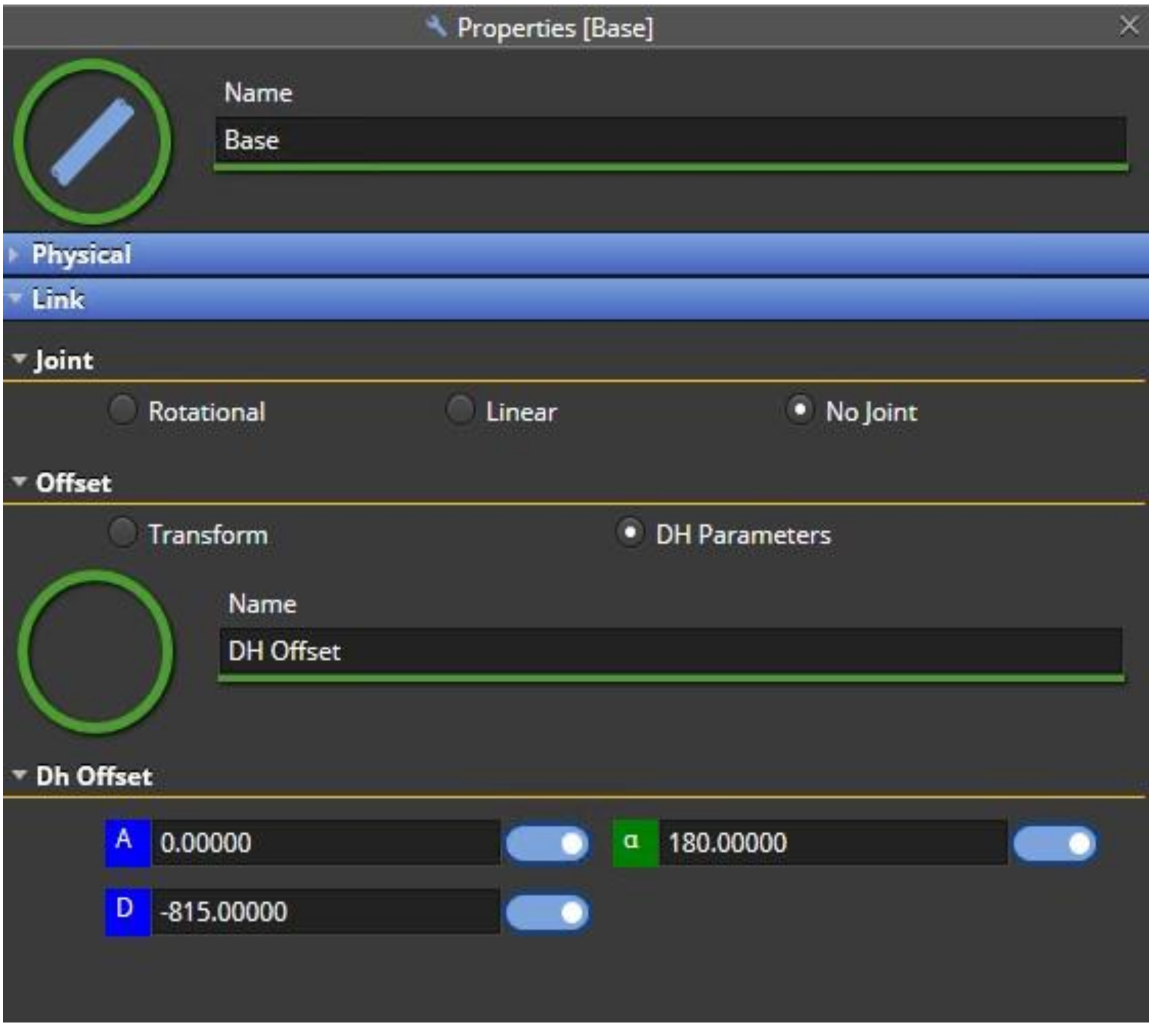
The following example will show how to model a Kuka robot manually.

Right click on **Machines** and select *New* | *Robots* | *Default 6 Axes*. A robot will be created with cylinders used as links.





The properties of each link can be changed as follows by defining the DH parameters.



Properties [Base]

Name: Base

Physical

Link

Joint

○ Rotational ○ Linear ● No Joint

Offset

○ Transform ● DH Parameters

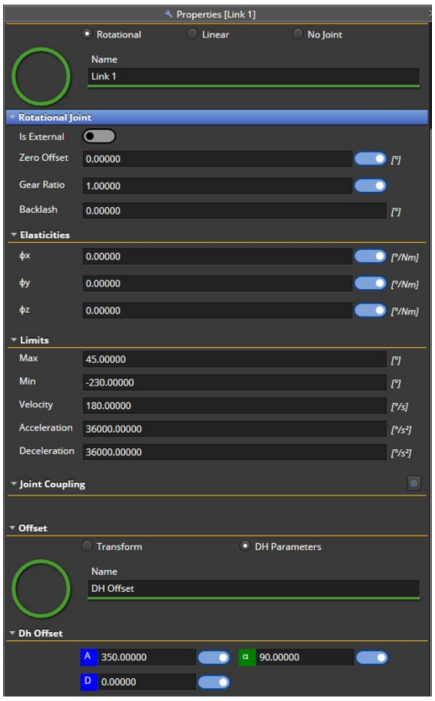
Name: DH Offset

Dh Offset

A: 0.00000

α: 180.00000

D: -815.00000



Properties [Link 1]

● Rotational ○ Linear ○ No Joint

Name: Link 1

Rotational Joint

Is External

Zero Offset 0.00000 [°]

Gear Ratio 1.00000

Backlash 0.00000 [°]

Elasticities

ϕx 0.00000 [°/Nm]

ϕy 0.00000 [°/Nm]

ϕz 0.00000 [°/Nm]

Limits

Max 45.00000 [°]

Min -230.00000 [°]

Velocity 180.00000 [°/s]

Acceleration 36000.00000 [°/s²]

Deceleration 36000.00000 [°/s²]

Joint Coupling

Offset

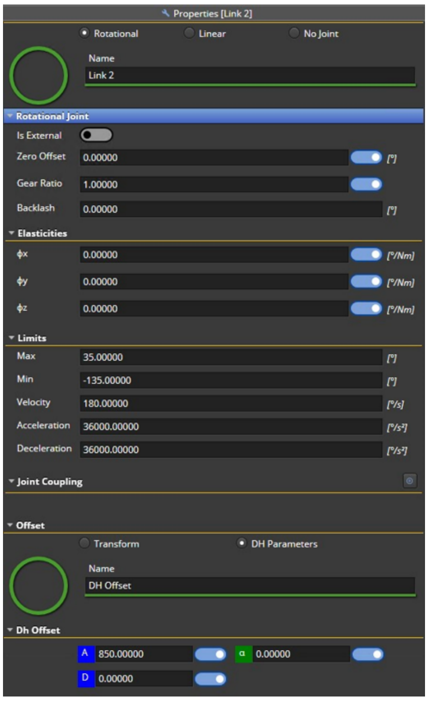
○ Transform ● DH Parameters

Name: DH Offset

Dh Offset

A 350.00000 α 90.00000

D 0.00000



Properties [Link 2]

● Rotational ○ Linear ○ No Joint

Name: Link 2

Rotational Joint

Is External

Zero Offset 0.00000 [°]

Gear Ratio 1.00000

Backlash 0.00000 [°]

Elasticities

ϕx 0.00000 [°/Nm]

ϕy 0.00000 [°/Nm]

ϕz 0.00000 [°/Nm]

Limits

Max 35.00000 [°]

Min -135.00000 [°]

Velocity 180.00000 [°/s]

Acceleration 36000.00000 [°/s²]

Deceleration 36000.00000 [°/s²]

Joint Coupling

Offset

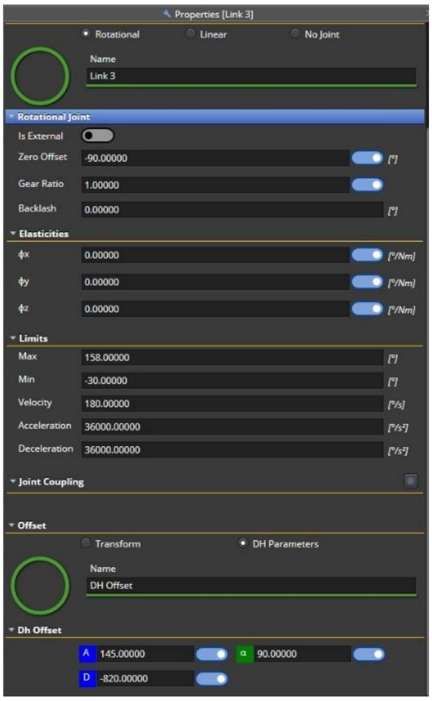
○ Transform ● DH Parameters

Name: DH Offset

Dh Offset

A 850.00000 α 0.00000

D 0.00000



Properties [Link 3]

● Rotational ○ Linear ○ No Joint

Name: Link 3

Rotational Joint

Is External

Zero Offset -90.00000 [°]

Gear Ratio 1.00000

Backlash 0.00000 [°]

Elasticities

ϕx 0.00000 [°/Nm]

ϕy 0.00000 [°/Nm]

ϕz 0.00000 [°/Nm]

Limits

Max 158.00000 [°]

Min -30.00000 [°]

Velocity 180.00000 [°/s]

Acceleration 36000.00000 [°/s²]

Deceleration 36000.00000 [°/s²]

Joint Coupling

Offset

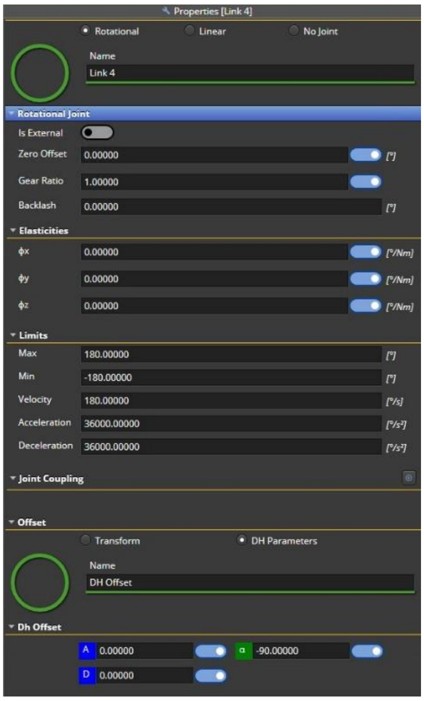
○ Transform ● DH Parameters

Name: DH Offset

Dh Offset

A 145.00000 α 90.00000

D -820.00000



Properties [Link 4]

● Rotational ○ Linear ○ No Joint

Name: Link 4

Rotational Joint

Is External

Zero Offset 0.00000 [°]

Gear Ratio 1.00000

Backlash 0.00000 [°]

Elasticities

ϕx 0.00000 [°/Nm]

ϕy 0.00000 [°/Nm]

ϕz 0.00000 [°/Nm]

Limits

Max 180.00000 [°]

Min -180.00000 [°]

Velocity 180.00000 [°/s]

Acceleration 36000.00000 [°/s²]

Deceleration 36000.00000 [°/s²]

Joint Coupling

Offset

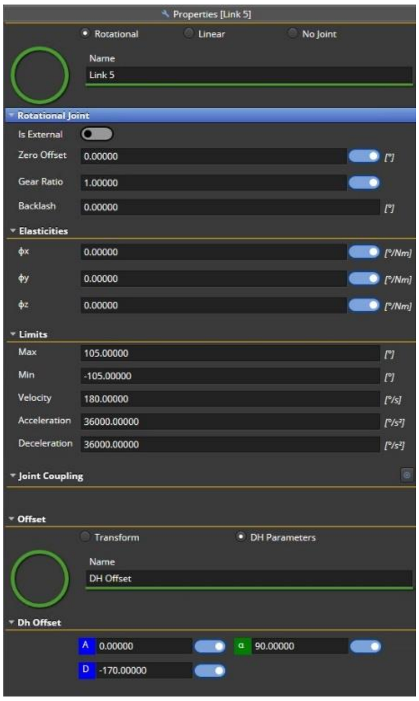
○ Transform ● DH Parameters

Name: DH Offset

Dh Offset

A 0.00000 α -90.00000

D 0.00000



Properties [Link 5]

● Rotational ○ Linear ○ No Joint

Name: Link 5

Rotational Joint

Is External

Zero Offset 0.00000 [°]

Gear Ratio 1.00000

Backlash 0.00000 [°]

Elasticities

ϕx 0.00000 [°/Nm]

ϕy 0.00000 [°/Nm]

ϕz 0.00000 [°/Nm]

Limits

Max 105.00000 [°]

Min -105.00000 [°]

Velocity 180.00000 [°/s]

Acceleration 36000.00000 [°/s²]

Deceleration 36000.00000 [°/s²]

Joint Coupling

Offset

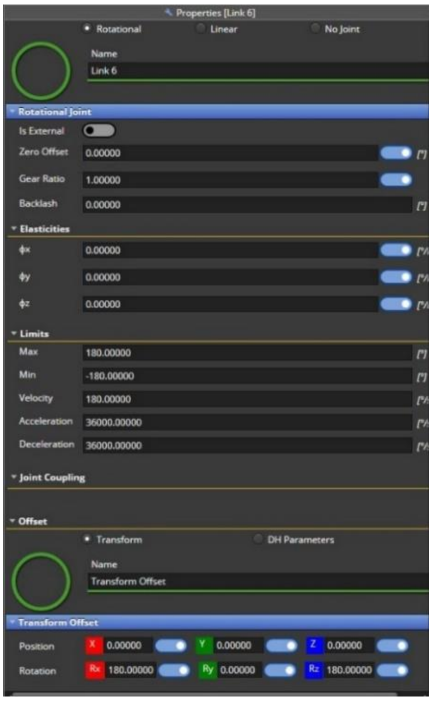
○ Transform ● DH Parameters

Name: DH Offset

Dh Offset

A 0.00000 α 90.00000

D -170.00000



Properties [Link 6]

● Rotational ○ Linear ○ No Joint

Name: Link 6

Rotational Joint

Is External

Zero Offset 0.00000 [°]

Gear Ratio 1.00000

Backlash 0.00000 [°]

Elasticities

ϕx 0.00000

ϕy 0.00000

ϕz 0.00000

Limits

Max 180.00000 [°]

Min -180.00000 [°]

Velocity 180.00000

Acceleration 36000.00000

Deceleration 36000.00000

Joint Coupling

Offset

● Transform ○ DH Parameters

Name: Transform Offset

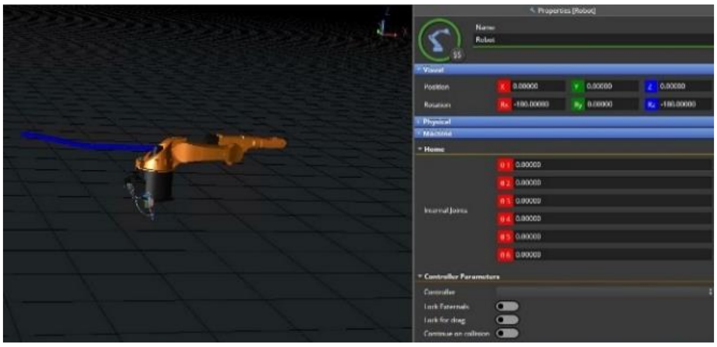
Transform Offset

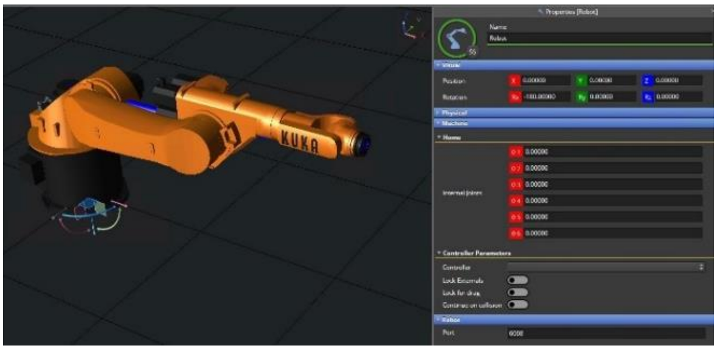
Position X 0.00000 Y 0.00000 Z 0.00000

Rotation Rx 180.00000 Ry 0.00000 Rz 180.00000

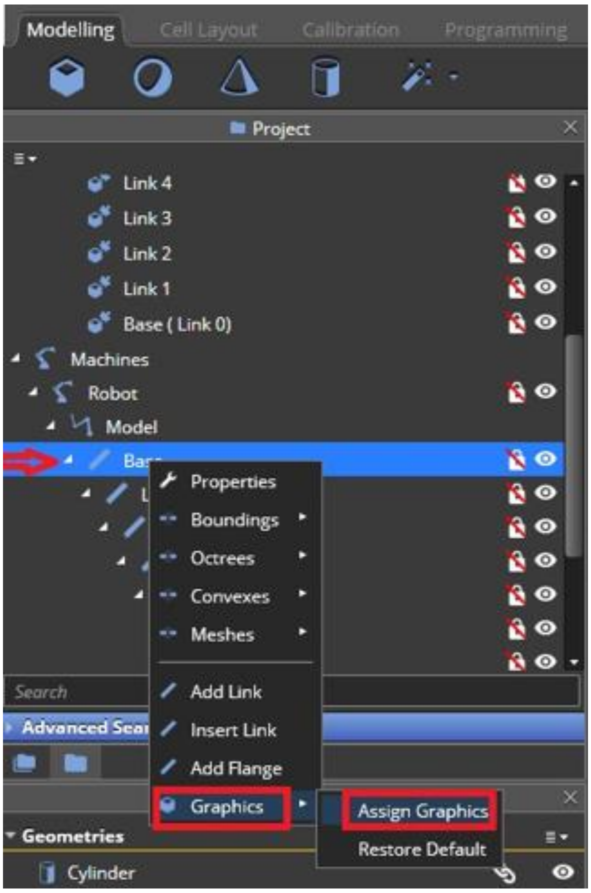
After finishing to edit the DH parameters, it is possible to assign the CAD files to the robot links.

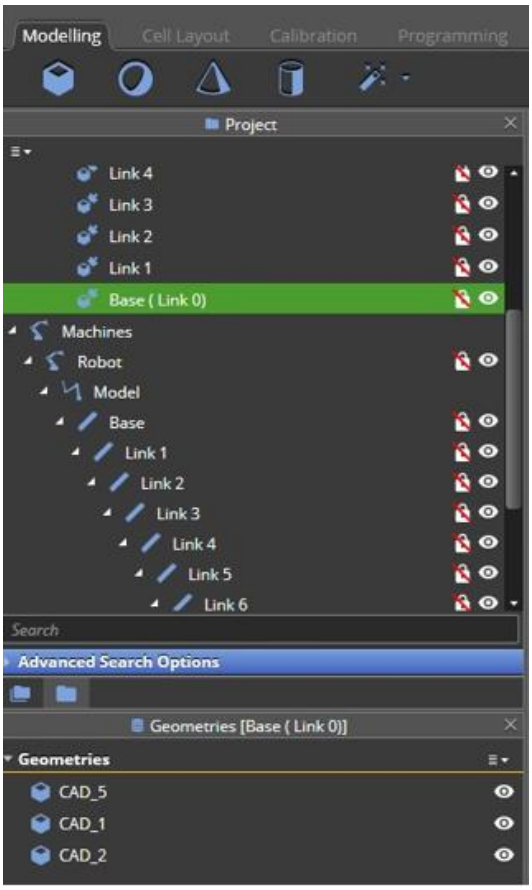
First it is needed to unhide the Robot CAD under Cell Items and change robot cylinders representation to match the CAD files by changing **R**$_{\mathrm{Z}}$ value from -180° to 0°.





To assign a graphics, right-click on the robot part and select *Graphics* | *Assign Graphics* and pick the related CAD item from the list.





5.1.3.1 Robot creation wizard

After clicking on the **Machine Creation** wizard, on the **Properties** window:

1) Select **Robot** as machine type, select **Create** as machine dimensions property and click **Add Joint** and select **No Joint** from the drop-down menu.

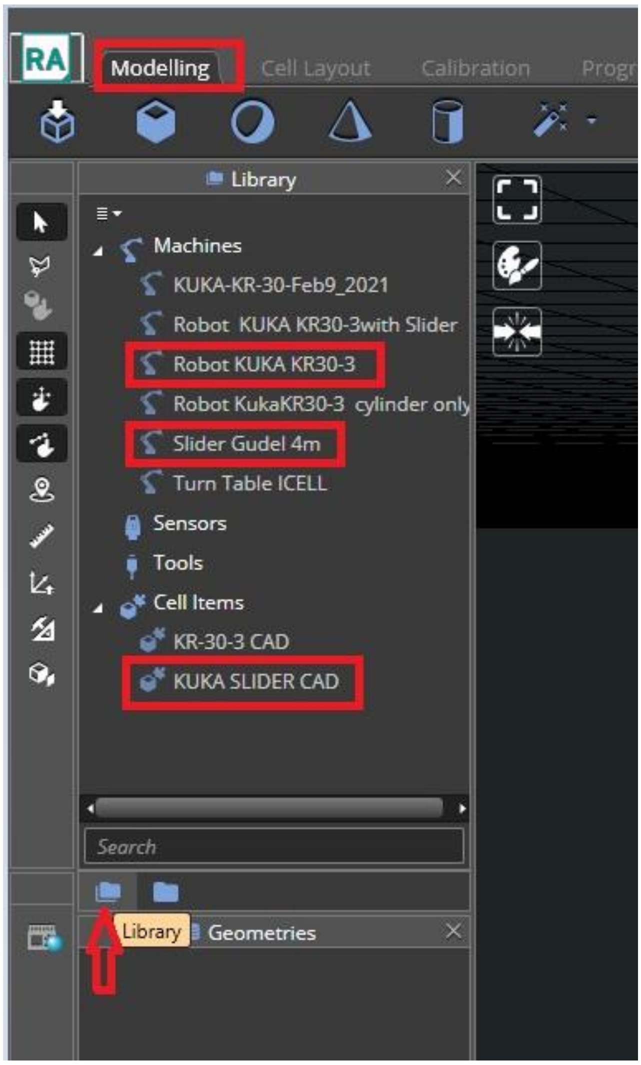
2) Add another 6 joints (Rotational) and click **Next**, leaving the internal joints parameter as default.

3) Continue as explained in 5.1.1.1 following the next steps of the wizard defining the minimum and maximum values of the joint limits and assigning the graphics (or skipping this part).

At the end of the process under **Machines** in the project tree you will find the new machine. If the graphics step was skipped, no graphics will be displayed.

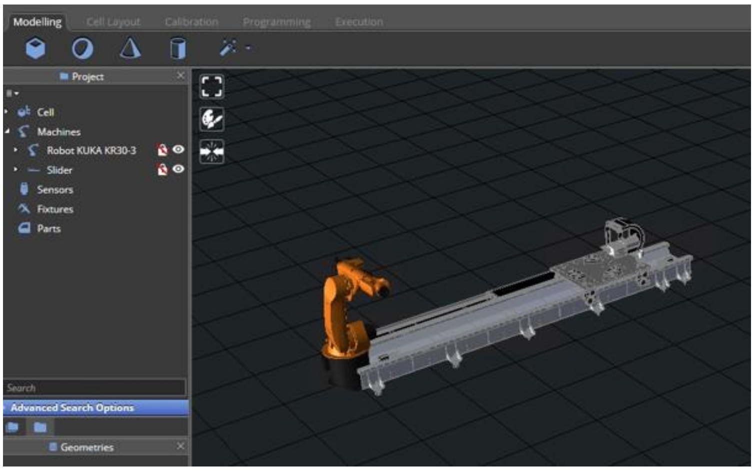
5.1.3.2 Robot on slider

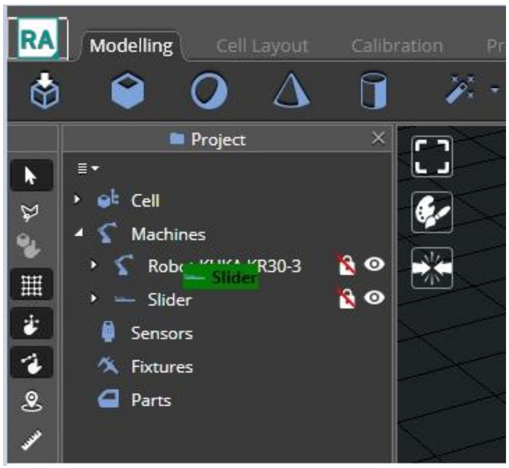
Make sure that you have the Machine model of the desired Slider and Robot in your library. Sometimes, the orientation of Robot and Slider CADs do not match, therefore you also need the Slider CAD in your library, separated already to Moving and Fixed groups as explained in chapter 5.1.1.



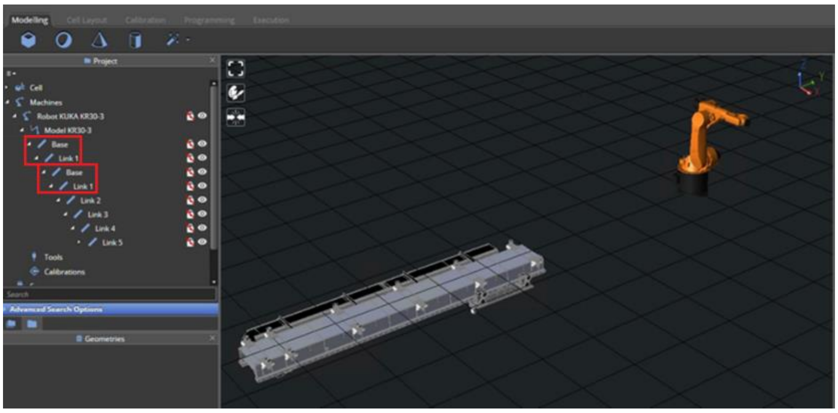
To combine the Robot and Slider into one Machine, drag and drop the desired robot from the library into the project. Select the robot, left click with mouse and drag it to the right. Repeat the same for the Slider (e.g., **Slider Gudel 4m** depicted above).
Within the project tree, under **Machines** you will have the robot and the slider.

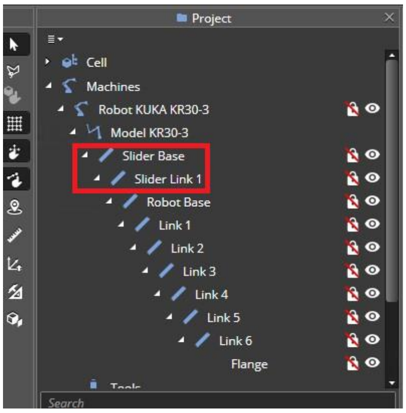
Select the slider with the mouse and drag it under the Robot. Once it is under the Robot, it will turn into green allowing you to drop it there.





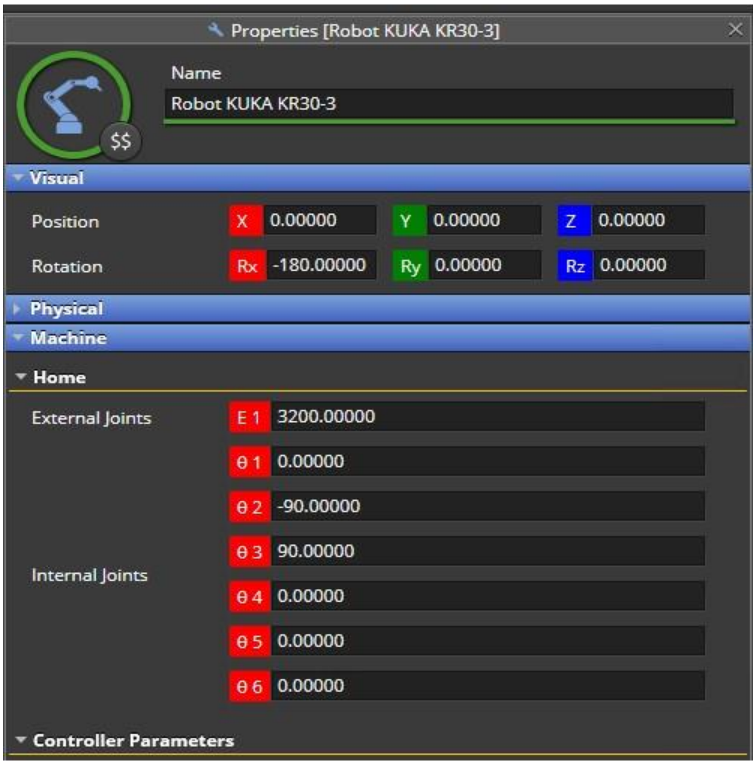
It is highly likely that the Robot model and Slider will not be in same coordinate system: you will notice there are two **Base** and two **Link1** items in the project tree as depicted in the picture below.
You can rename the slider related items into **Slider Base** and **Slider Link 1**.





5.1.3.3 Align Robot and Slider orientation and Position.

Select the robot properties and set the robot **Home** values for the slider (**E1**) and robot Joints (**θ1-θ6**) as they are in the real cell.



Properties [Robot KUKA KR30-3]

Name: Robot KUKA KR30-3

Visual

Position	X 0.00000	Y 0.00000	Z 0.00000
Rotation	Rx -180.00000	Ry 0.00000	Rz 0.00000

Physical

Machine

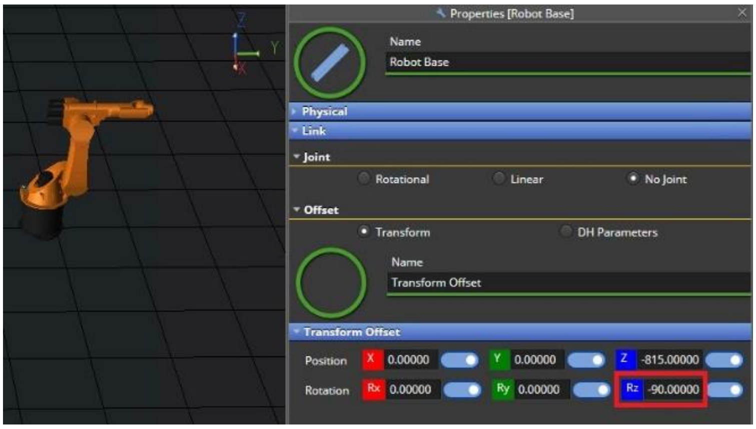
Home

External Joints	E 1	3200.00000
Internal Joints	θ 1	0.00000
	θ 2	-90.00000
	θ 3	90.00000
	θ 4	0.00000
	θ 5	0.00000
	θ 6	0.00000

Controller Parameters

The next step is to correct the Robot Orientation in its home (we need to Rotate it 90° counter clockwise) so in robot **Base** properties it is needed to change the **Offset** parameter from **DH parameters** to **Transform**: The **R**$_Z$ must be changed to -90°.

Make sure also to change the robot **Base** CAD accordingly.



Properties [Robot Base]

Name: Robot Base

Physical

Link

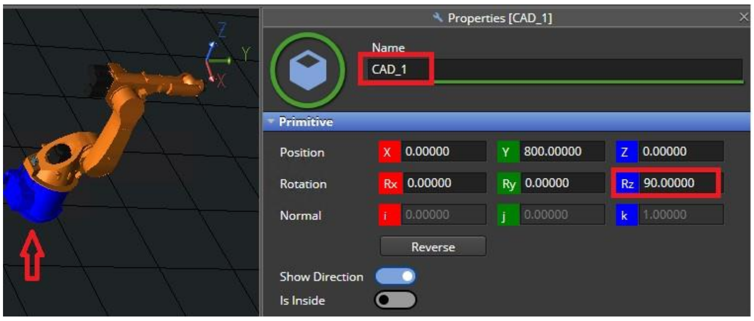
Joint: ○ Rotational ○ Linear ● No Joint

Offset: ● Transform ○ DH Parameters

Name: Transform Offset

Transform Offset

Position	X 0.00000	Y 0.00000	Z -815.00000
Rotation	Rx 0.00000	Ry 0.00000	Rz -90.00000



Properties [CAD_1]

Name: CAD_1

Primitive

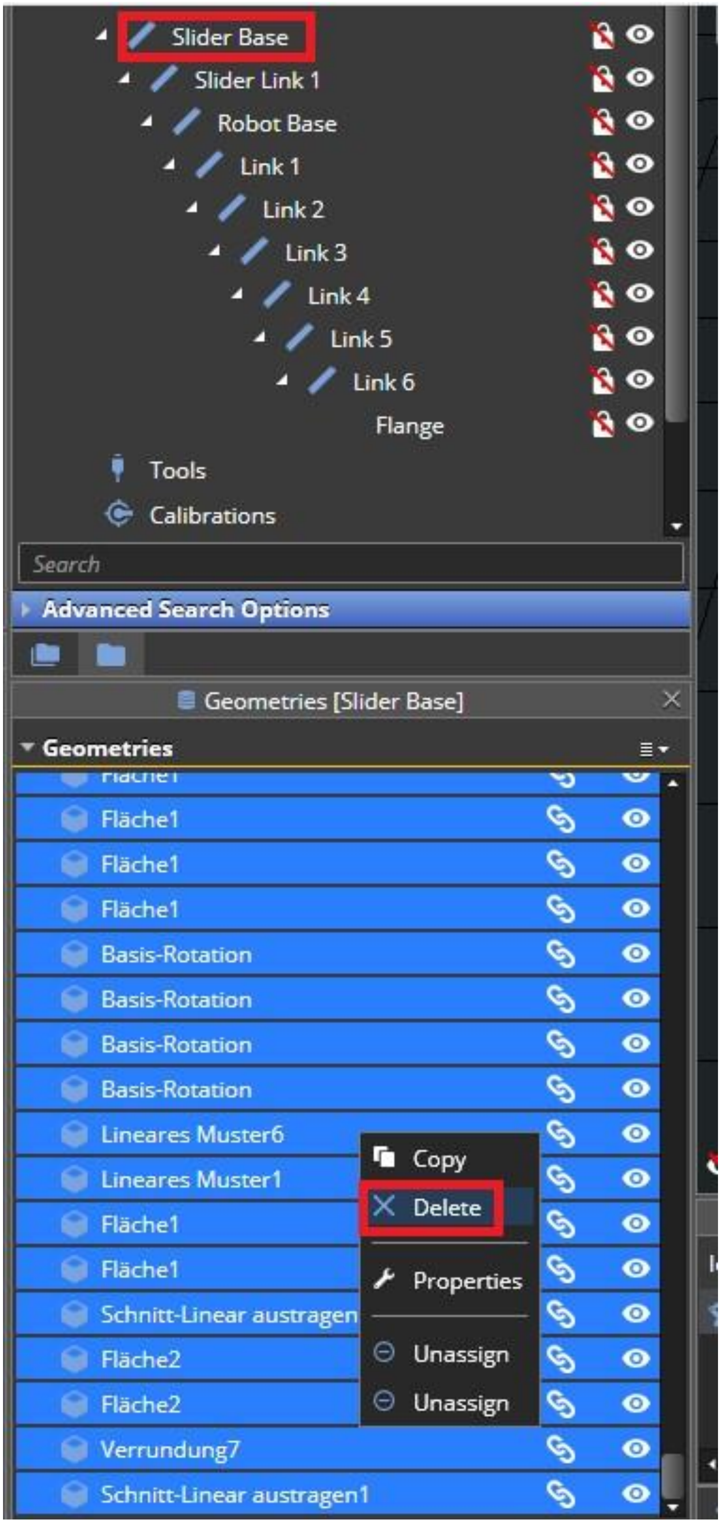
Position	X 0.00000	Y 800.00000	Z 0.00000
Rotation	Rx 0.00000	Ry 0.00000	Rz 90.00000
Normal	i 0.00000	j 0.00000	k 1.00000

Reverse

Show Direction: on

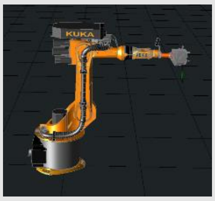
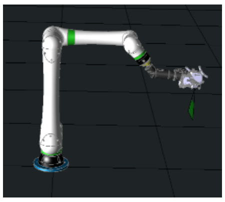
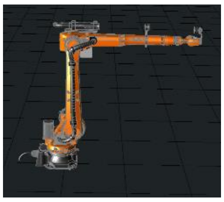
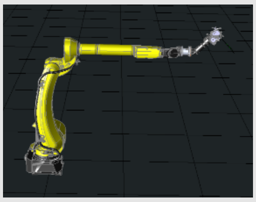
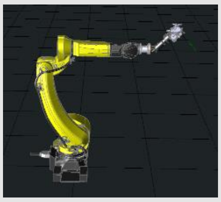
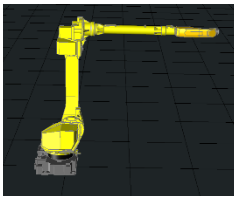
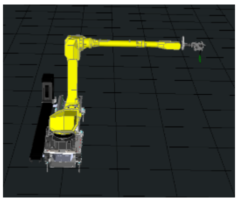
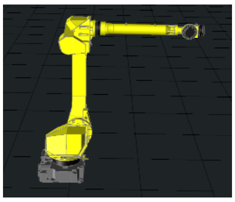
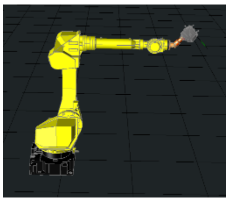
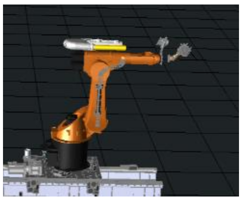
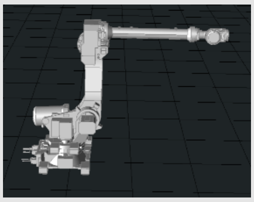
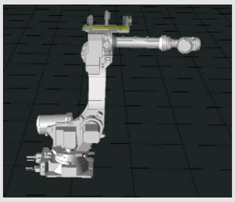
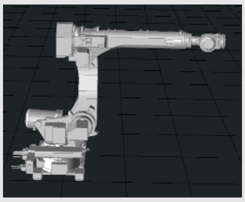
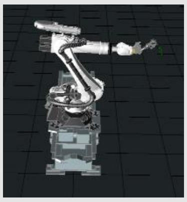
Is Inside: off

The next step is to remove the slider assigned graphic CADs which are in wrong orientation and the same is to be repeated also for slider **Link 1**.



The next step is to align the slider CAD group in its properties under the robot **Base**. Before you do that, make sure you have the correct slider **E1** values as its position. In our example it is in 4000 mm.

5.1.3.4 List of supported robot models

Fanuc robot model using R-30ia controller	**KUKA robot model** using KR C4 controller
LR MATE 200iD-7C 	KR 6 2 
CRX-10iA/L 	KR 20 R3100 
M-20iD/12L  M-20iD/25 	KR 22 R1610
M-710iC/12L  M-710iC/20L  M-710iC/45  M-710iC/50 	KR 30-3 
R-2000iC/125L  R-2000iC/165F  R-2000iC/210L 	KR 90 R2700 Pro 
	KR 210 R2700-2

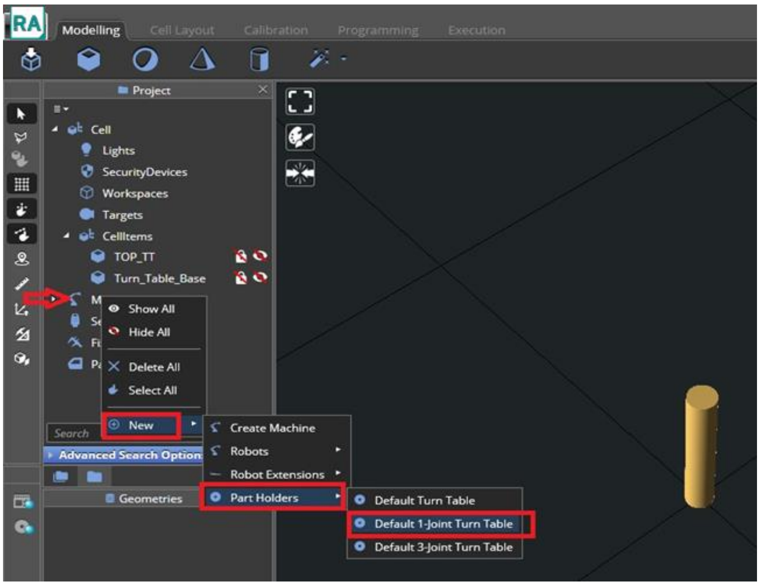
5.1.4 Turn table

Following the same steps explained in chapter 5.1.1 it is possible to import the Turn Table's CAD files. The CAD will show under **Cell Items** as a **Group**. The CAD elements can be separated and merged into two groups: **Turn_Table_Base** (fixed), and **Top_TT** (rotating).

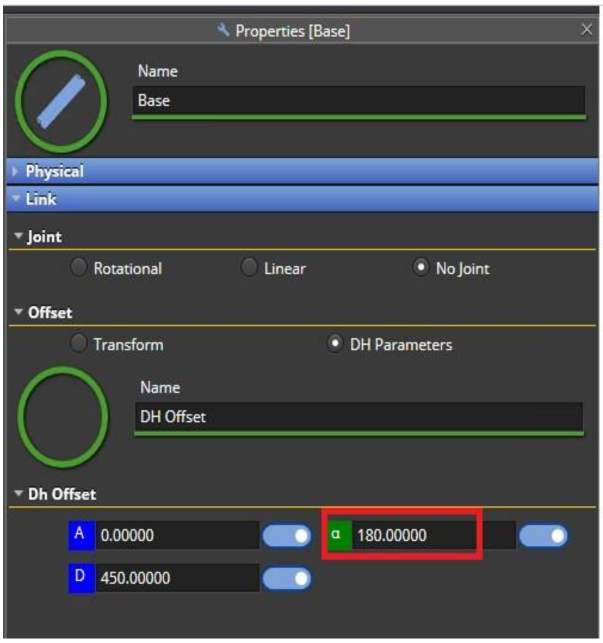
It may be needed to change the CAD properties to make sure Turn Table Cad orientation facing up to Z axis.

A Turn Table can be created manually of following the Turn Table creation wizard.

Under Cell items group, right click on **Machines**, then select *New* | *Part Holders* | *Default 1-Joint Turn Table*. A cylinder will be created.



On the cylinder properties, depending on the rotation direction, it is possible to set the correct DH parameters, e.g., changing the **α** value to 180° will make the cylinder rotating in positive direction in clockwise direction.


Properties [Base]

Name: Base

Physical

Link

Joint: ○ Rotational ○ Linear ● No Joint

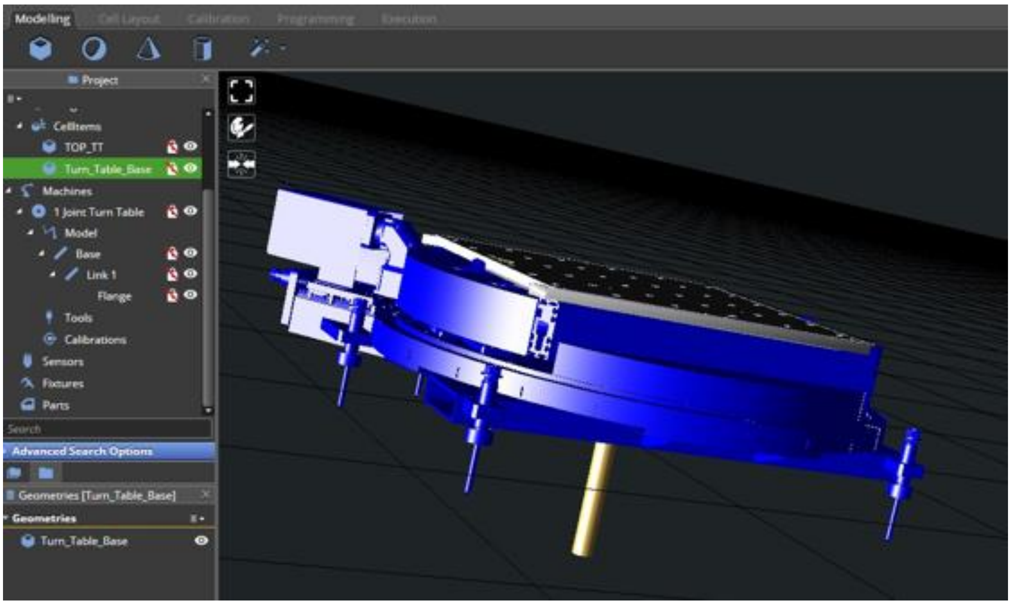
Offset: ○ Transform ● DH Parameters

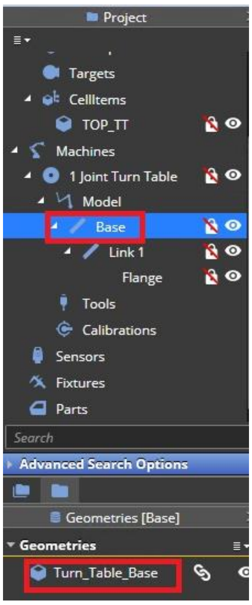
Name: DH Offset

Dh Offset

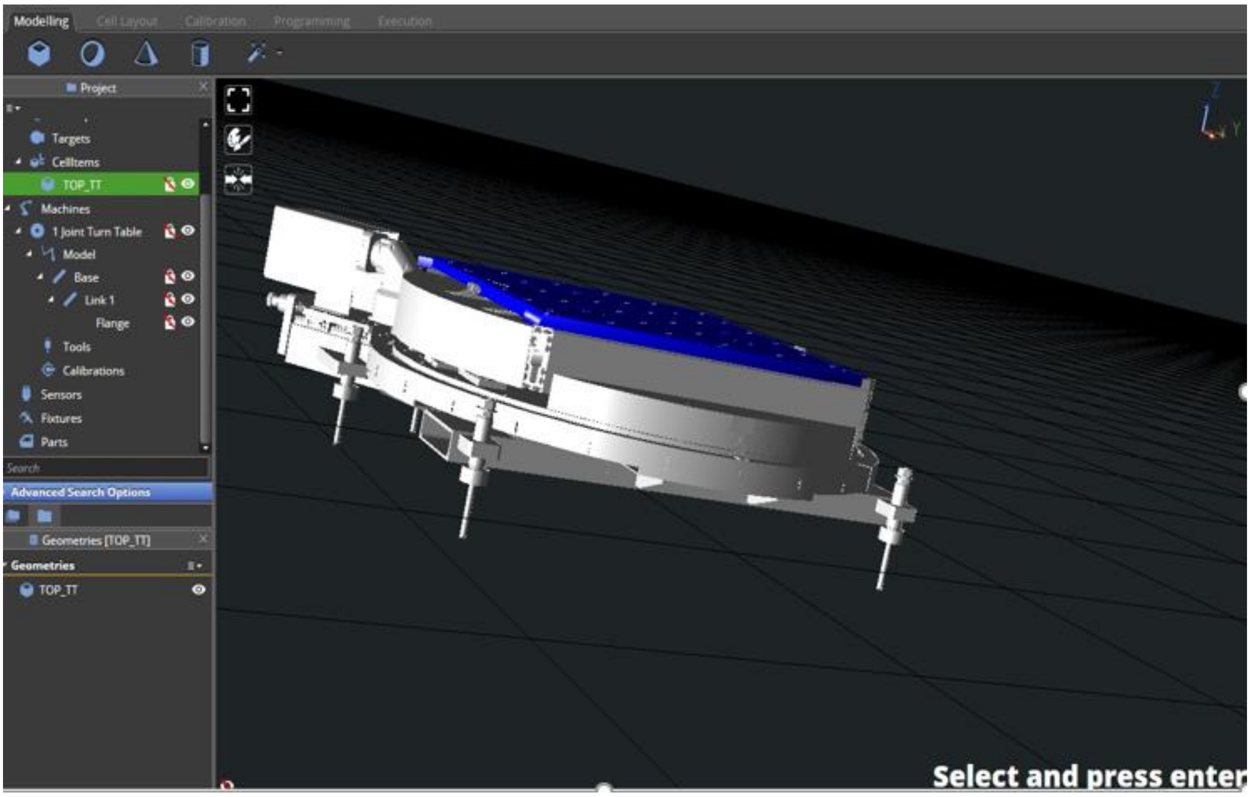
A 0.00000
α 180.00000
D 450.00000

Now it is possible to assign the graphics to **Base** and **Link 1**. By right-clicking on **Base** and selecting *Graphics* | *Assign Graphics* and selecting with the mouse the Turn Table CAD, it will turn to green. After pressing the [ENTER] key, clicking on the **Base** again, you will notice that the **Geometries** will include the Turn table base CAD that will no longer be under **Cell Items** in the project tree.





The same can be done for **Link1** and **TOP_TT** CAD can be assigned.



Select and press enter

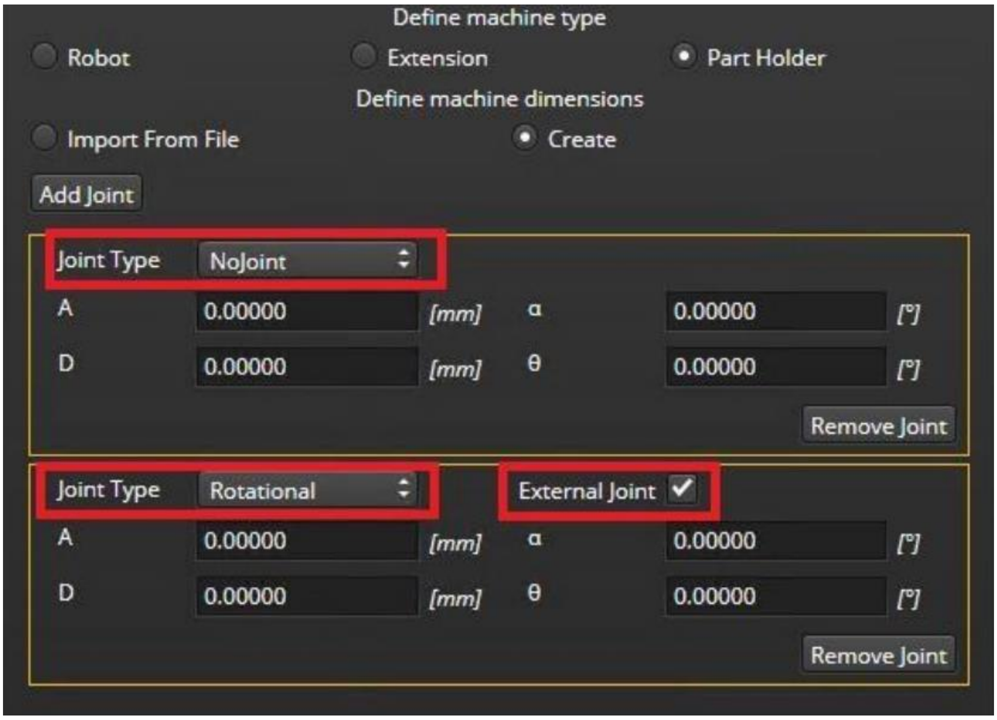
5.1.4.1 Turn Table creation wizard

The turn table creation wizard reflects the same steps explained in chapter 5.1.1.1, but in this case the second joint to be added will be a Rotational joint.

After clicking on the **Machine Creation** wizard, on the **Properties** window:

1) Select **Part Holder** as machine type, select **Create** as machine dimensions, and click **Add Joint** and select **No Joint** from the **Joint Type** drop-down menu.

2) Add another joint selecting **Rotational** for the moving joint and checking **External Joint**, leaving the D-H parameters unchanged. Once done, click on the arrow on bottom right, to continue to the next step.



Define machine type

○ Robot ○ Extension ● Part Holder

Define machine dimensions

○ Import From File ● Create

Add Joint

Joint Type: NoJoint

A	0.00000	[mm]	α	0.00000	[°]
D	0.00000	[mm]	θ	0.00000	[°]

Remove Joint

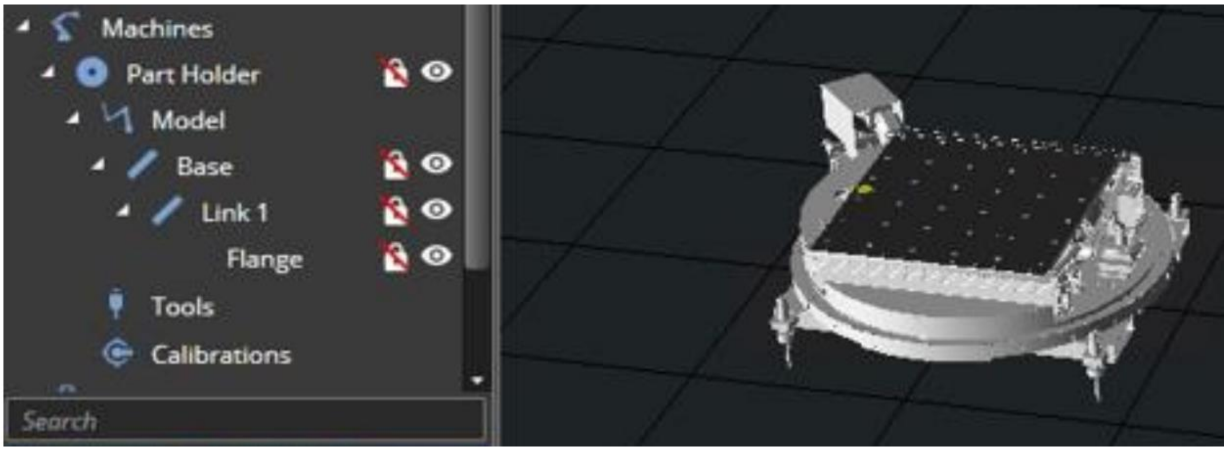
Joint Type: Rotational — External Joint ☑

A	0.00000	[mm]	α	0.00000	[°]
D	0.00000	[mm]	θ	0.00000	[°]

Remove Joint

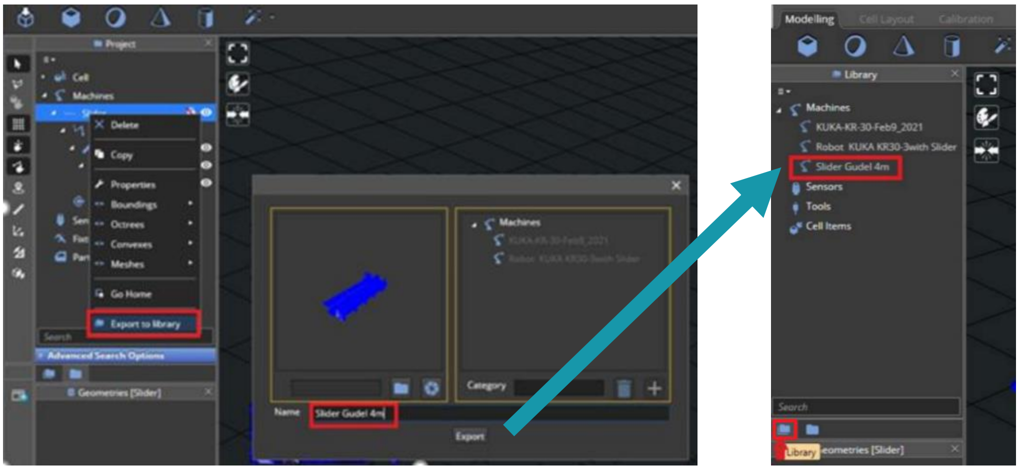
3) Continue as explained in 5.1.1.1 following the next steps of the wizard defining the minimum and maximum values of the joint limits and assigning the graphics.

At the end of the process under **Machines** in the project tree you will find the new machine.



5.1.5 Exporting an object to the library

Select the object to be exported, right-click and select **Export to library**. A dialog window will appear where it is possible to specify the object **Category** and **Name**. After clicking **Export**, the object will be added to the **Library** for future use.

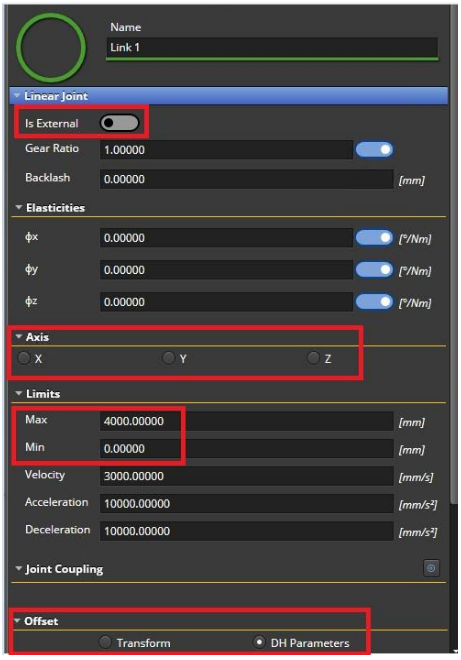


Name: Slider Gudel 4m

5.1.6 Changing the machine properties

In the project go to cell item, right click, and check the properties.
Depending on the machine type, different parameters can be changed.
E.g., considering the slider modelled in chapter 5.1.1 the default values must be changed to:

1) For the **Link 1**:
   a) **Linear Joint** should change to **Is External**
   b) **Axis** to be check to Y (in our example)
   c) **Offset** change to Transform Limits



Name: Link 1

Linear Joint
Is External: off
Gear Ratio: 1.00000
Backlash: 0.00000 [mm]

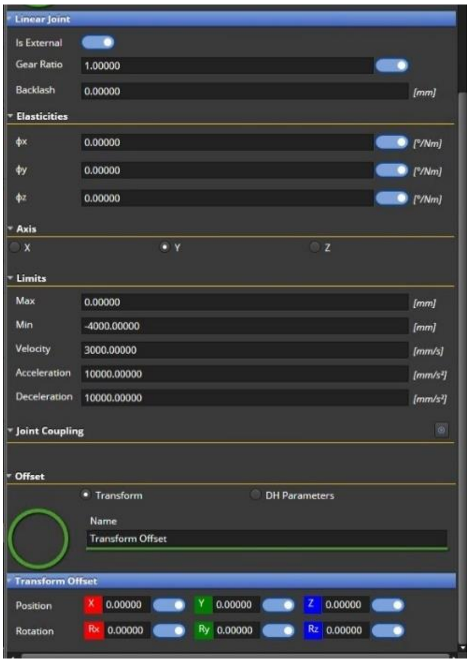
Elasticities
ϕx: 0.00000 [°/Nm]
ϕy: 0.00000 [°/Nm]
ϕz: 0.00000 [°/Nm]

Axis
○ X ○ Y ○ Z

Limits
Max: 4000.00000 [mm]
Min: 0.00000 [mm]
Velocity: 3000.00000 [mm/s]
Acceleration: 10000.00000 [mm/s²]
Deceleration: 10000.00000 [mm/s²]

Joint Coupling

Offset
○ Transform ● DH Parameters



Linear Joint
Is External: on
Gear Ratio: 1.00000
Backlash: 0.00000 [mm]

Elasticities
ϕx: 0.00000 [°/Nm]
ϕy: 0.00000 [°/Nm]
ϕz: 0.00000 [°/Nm]

Axis
○ X ● Y ○ Z

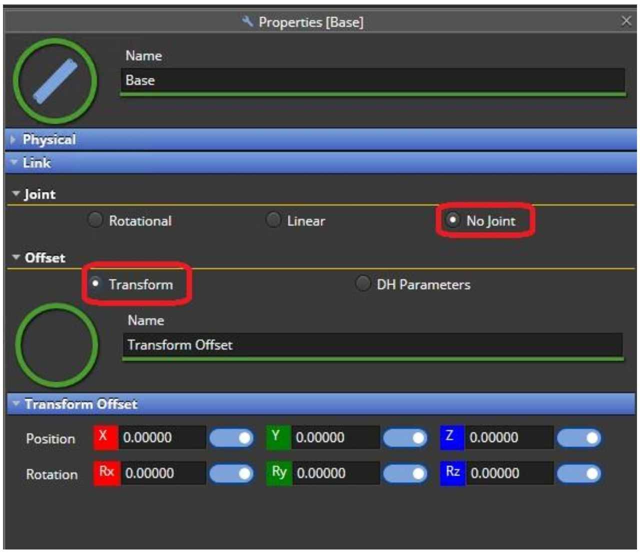
Limits
Max: 0.00000 [mm]
Min: -4000.00000 [mm]
Velocity: 3000.00000 [mm/s]
Acceleration: 10000.00000 [mm/s²]
Deceleration: 10000.00000 [mm/s²]

Joint Coupling

Offset
● Transform ○ DH Parameters
Name: Transform Offset

Transform Offset
Position: X 0.00000 Y 0.00000 Z 0.00000
Rotation: Rx 0.00000 Ry 0.00000 Rz 0.00000

2) For the **Base** verify that Slider Base is set to **No Joint** and **Offset** to **Transform**.



Properties [Base]

Name: Base

Physical

Link

Joint

○ Rotational ○ Linear ● No Joint

Offset

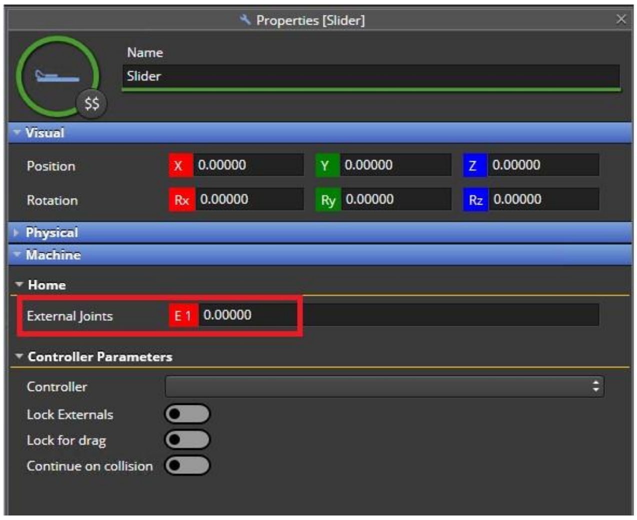
● Transform ○ DH Parameters

Name: Transform Offset

Transform Offset

Position	X	0.00000	Y	0.00000	Z	0.00000
Rotation	Rx	0.00000	Ry	0.00000	Rz	0.00000

3) Set **Home** position.



Properties [Slider]

Name: Slider

Visual

Position	X	0.00000	Y	0.00000	Z	0.00000
Rotation	Rx	0.00000	Ry	0.00000	Rz	0.00000

Physical

Machine

Home

External Joints E 1 0.00000

Controller Parameters

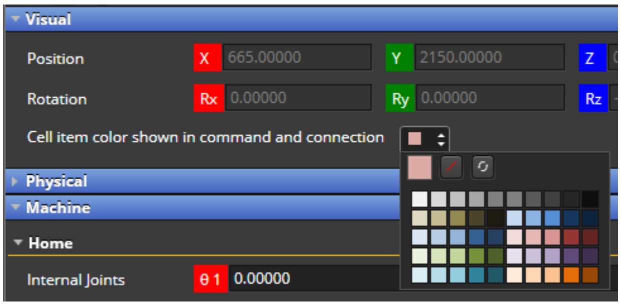
Controller

Lock Externals

Lock for drag

Continue on collision

4) Assign a colour to each cell device within the **Visual** field by picking a colour from the available ones or by specifying the RGB or Hex code.



Visual

Position	X	665.00000	Y	2150.00000	Z
Rotation	Rx	0.00000	Ry	0.00000	Rz

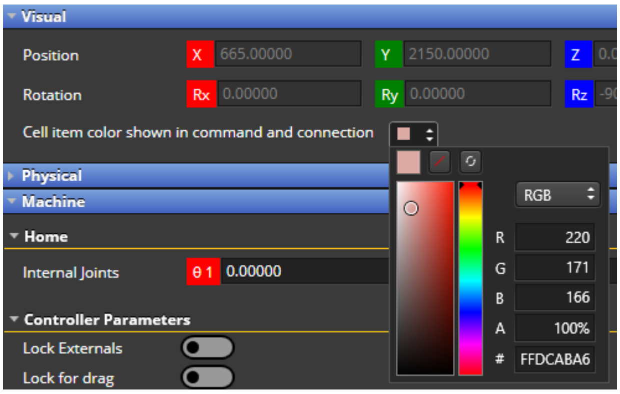
Cell item color shown in command and connection

Physical

Machine

Home

Internal Joints θ 1 0.00000



Visual

Position	X	665.00000	Y	2150.00000	Z	0.0
Rotation	Rx	0.00000	Ry	0.00000	Rz	-9

Cell item color shown in command and connection

Physical

Machine

Home

Internal Joints θ 1 0.00000

Controller Parameters

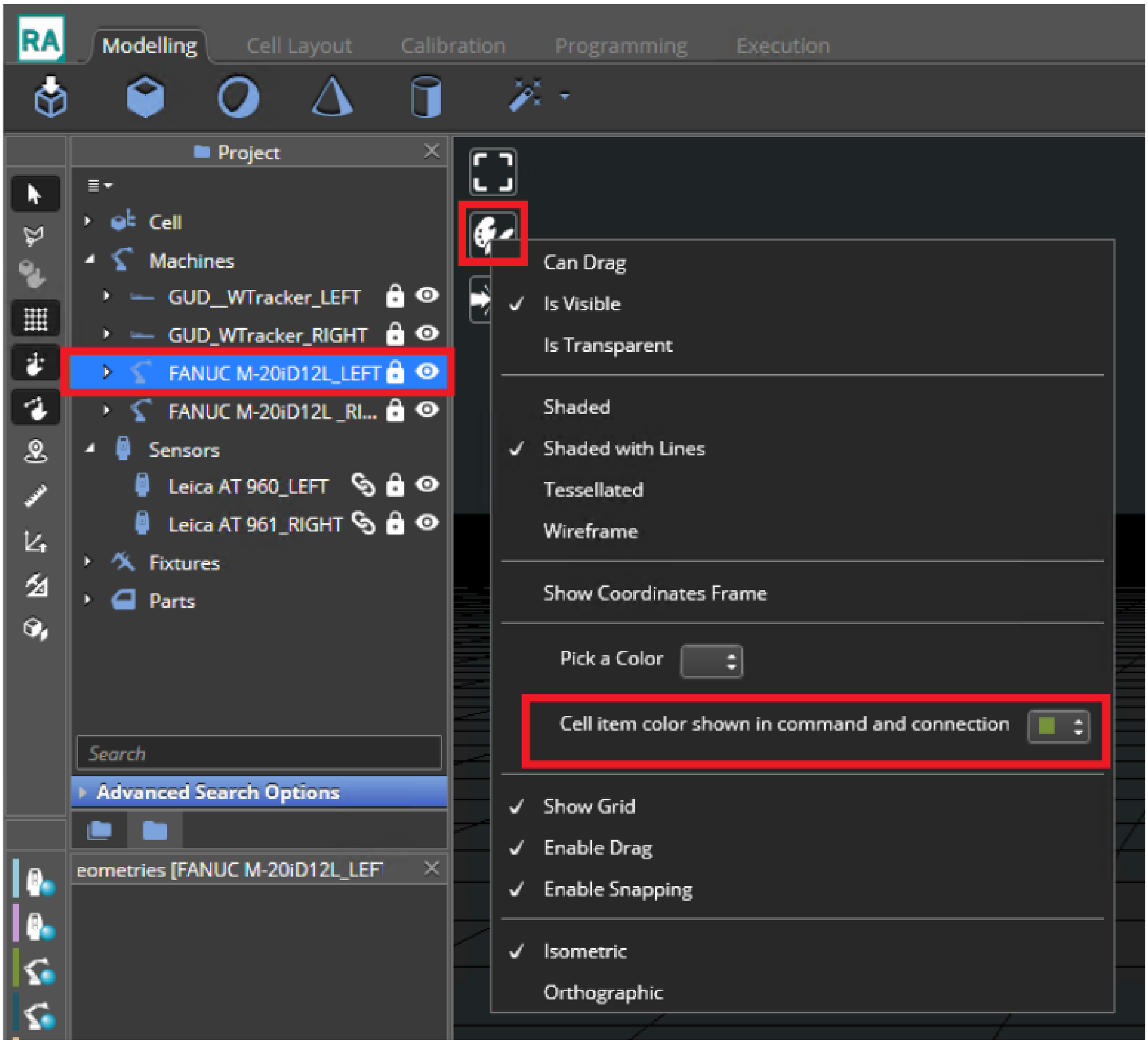
Lock Externals

Lock for drag

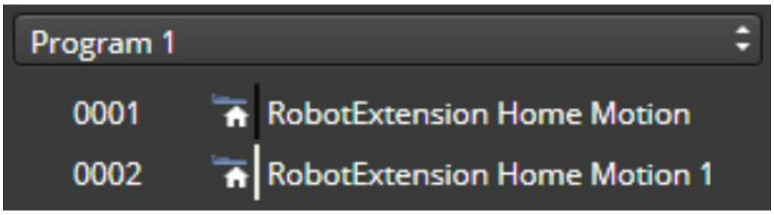
RGB

R	220
G	171
B	166
A	100%
#	FFDCABA6

It's also possible to select the cell item in the 3D Window, first right-clicking on the item and then selecting the colour from the drop-down menu by clicking on **Cell item colour shown in command and connection**.



By assigning a specific colour to all the cell items, it will be easier for the user to easily distinguish between which cell items are involved in which command of a program. E.g., in the picture below, two different sliders can be easily distinguished because one is depicted in black and the other one in white.


Program 1

0001 RobotExtension Home Motion

0002 RobotExtension Home Motion 1

5.1.7 Configuring machine saved positions

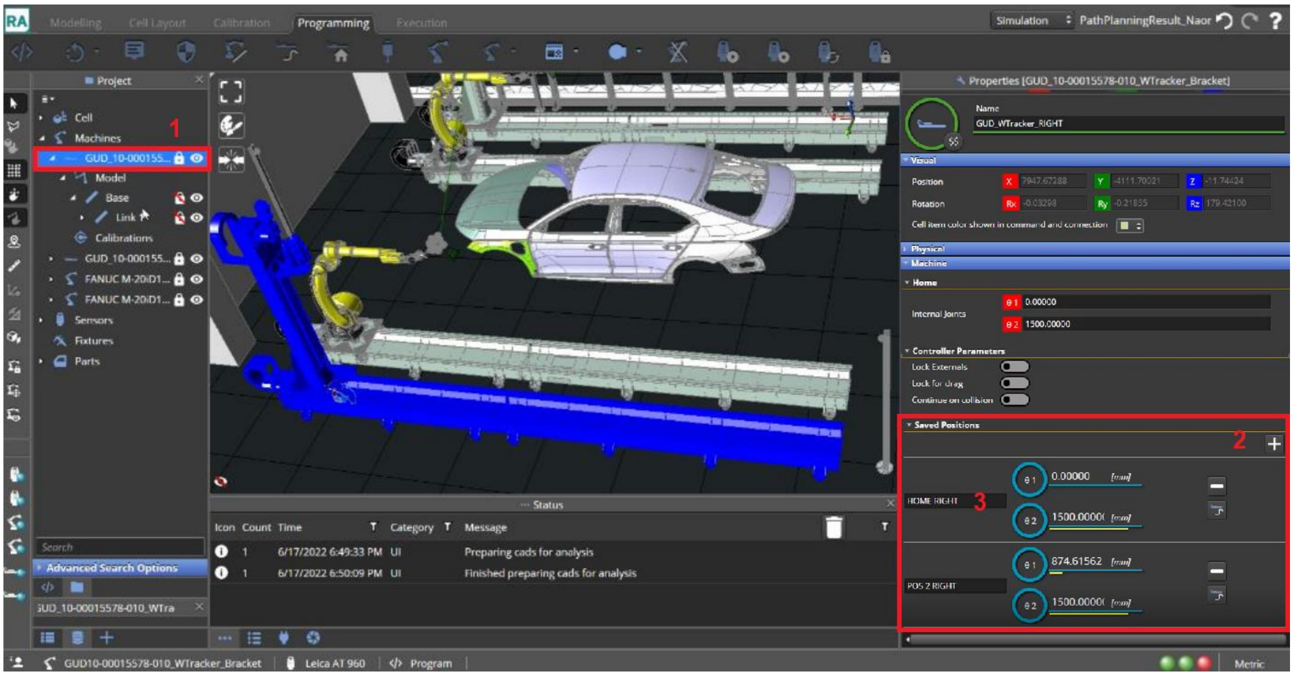
It is also possible to save some relevant positions to be used by the automatic path planning algorithm (see chapter 8.7 for more details) to move the machine to one of the saved positions to perform a proper scan, e.g., measuring the back of a door mounted on a fixture could require rotating the turn table by 180°.

This would reduce the effort of the path planning algorithm because the sampling would be done among the saved positions instead of the infinite positions available within the complete joint space of the machine, e.g., 360° for a turn table or 4m for a 4m-slider.

These positions are stored under **Saved Positions** property.

To save a position:

1) Move the machine in the desired position.
2) Click the plus button (⊞) to store it. The minus button (⊟) will remove the corresponding position.
3) Rename the position conveniently.



Properties [GUD_10-00015578-010_WTracker_Bracket]

Name: GUD_WTracker_RIGHT

Visual

Position	X 7947.67288	Y -4111.70021	Z -11.74424
Rotation	Rx -0.03298	Ry -0.21835	Rz 179.42100

Cell item color shown in command and connection

Physical

Machine

Home

Internal Joints: θ 1 0.00000; θ 2 1500.00000

Controller Parameters

Lock Externals
Lock for drag
Continue on collision

Saved Positions

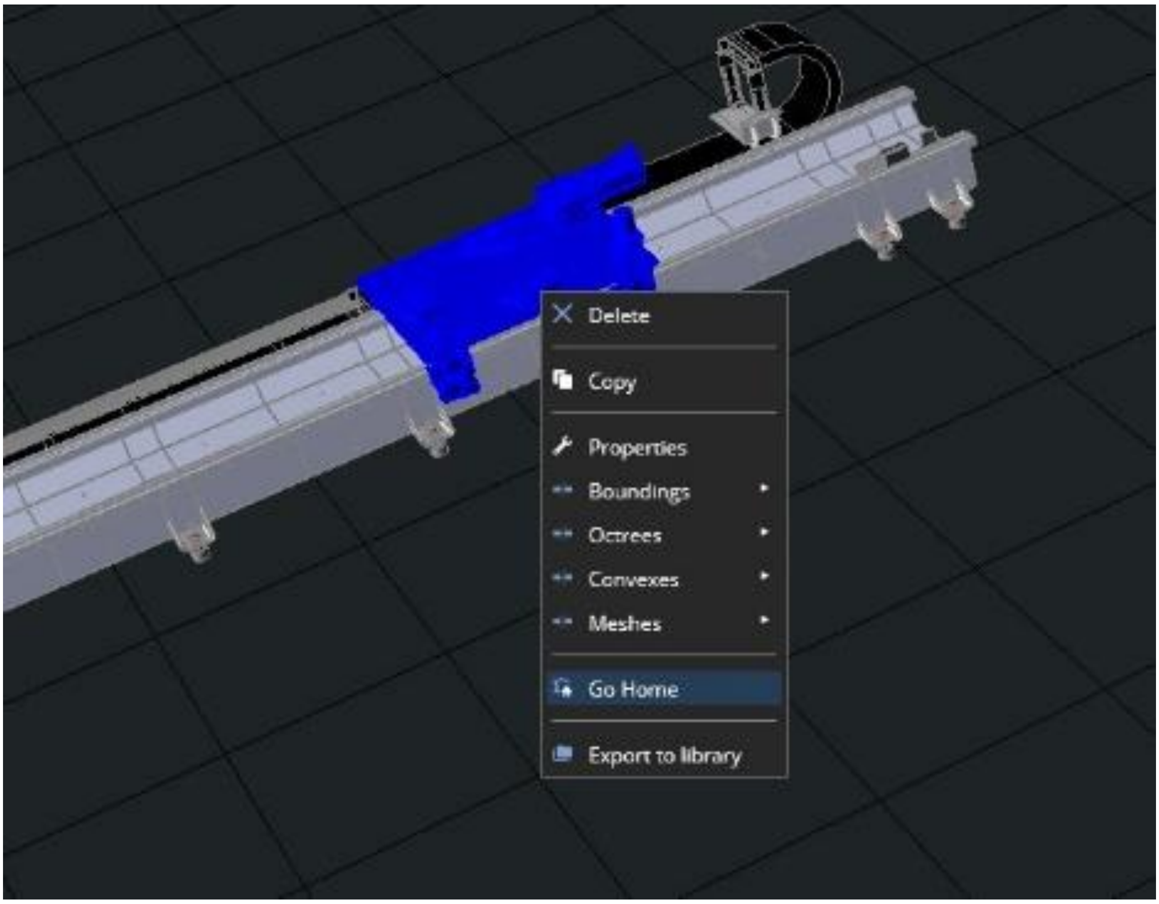
HOME RIGHT	θ 1 0.00000 [mm]	θ 2 1500.0000([mm]
POS 2 RIGHT	θ 1 874.61562 [mm]	θ 2 1500.0000([mm]

Status

Icon	Count	Time	Category	Message
ⓘ	1	6/17/2022 6:49:33 PM	UI	Preparing cads for analysis
ⓘ	1	6/17/2022 6:50:09 PM	UI	Finished preparing cads for analysis

5.1.8 Machine testing

Please refer to chapter 8.10 for more information on how to move the machines within the 3D environment to test the correct movements of the machines (slider linear movement, turn table rotational movement, robot cartesian or joints movements. To send the machine to home position it’s just needed to right-click on the machine in the 3D View and click **Go Home**.

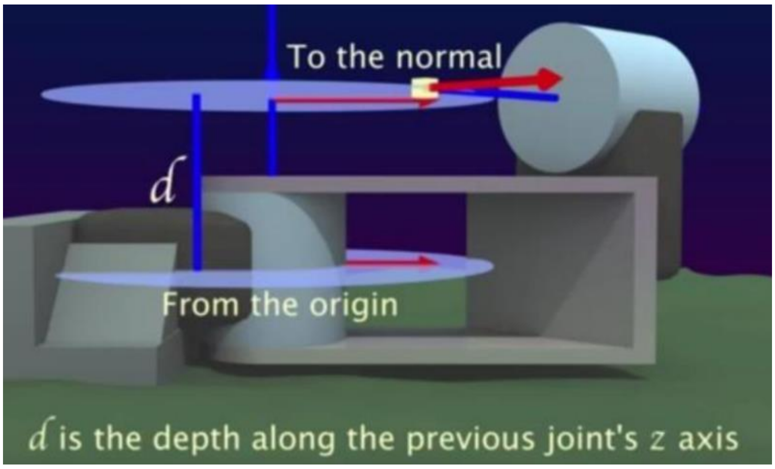


5.1.9 D-H parameters

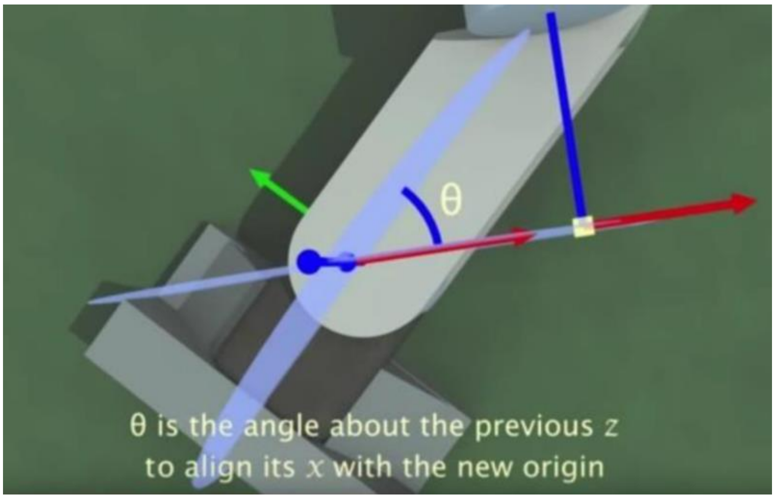
D-H (Denavit-Hartenberg) parameters are the four parameters associated with a particular convention for attaching reference frames to the links of a spatial kinematic chain or robot.

The D-H parameters are the following:

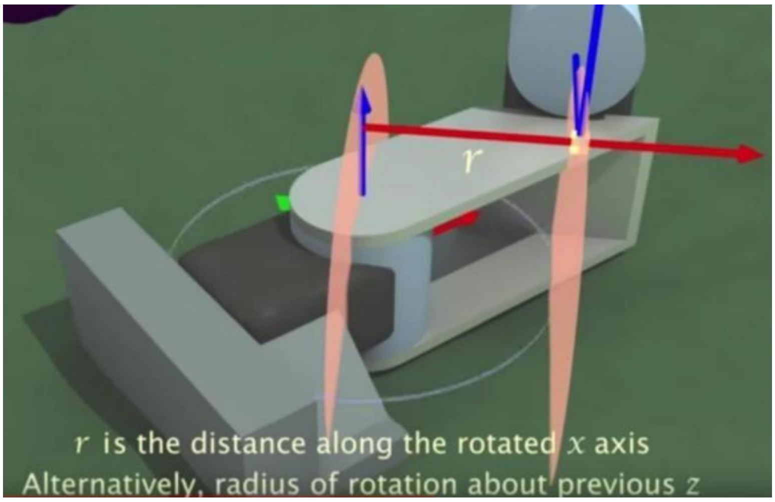
1) **D**: the depth along the earlier joint’s Z axis



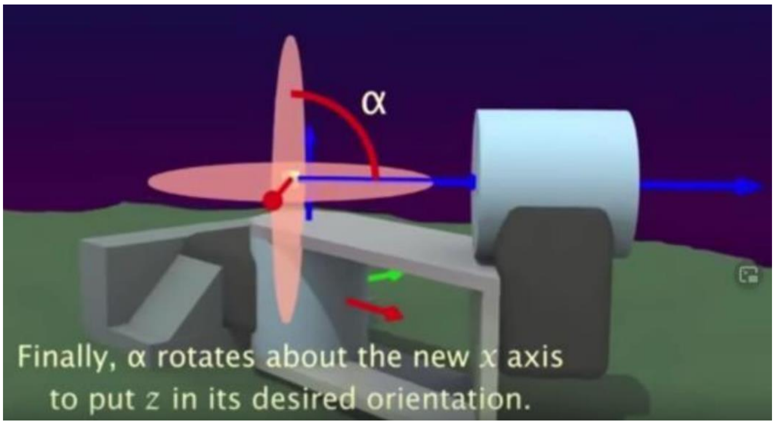
2) **θ**: the angle about the earlier Z axis to align its X with new origin. In HxGN Robotic Automation this is called **Zero offset**.


θ
θ is the angle about the previous z
to align its x with the new origin

3) **R**: the distance along the rotated x axis. In HxGN Robotic Automation this is called **A**.

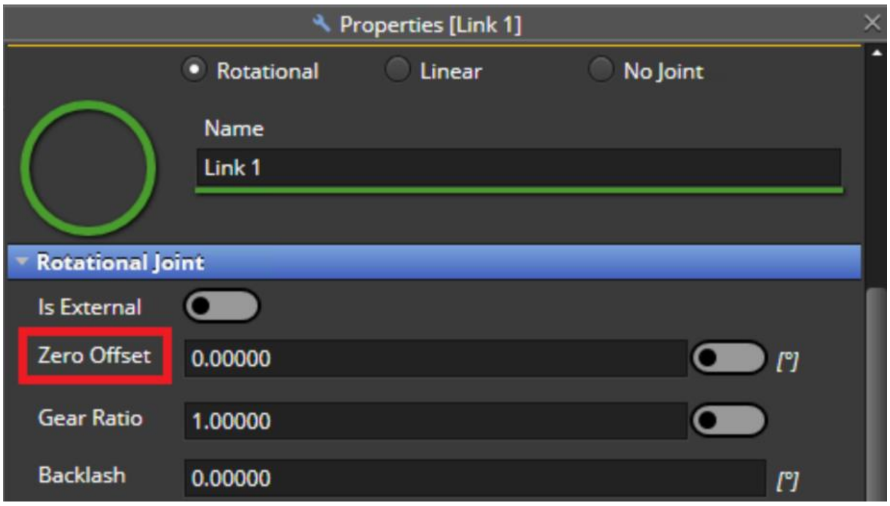

r
r is the distance along the rotated x axis
Alternatively, radius of rotation about previous z

4) **α**: the rotation angle about the new x axis to put Z in its desired orientation.


α
Finally, α rotates about the new x axis
to put z in its desired orientation.

Example – Properties of Link 1 of Fanuc robot

θ is configured as **Zero Offset** under the **Rotational Joint** properties:



Properties [Link 1]

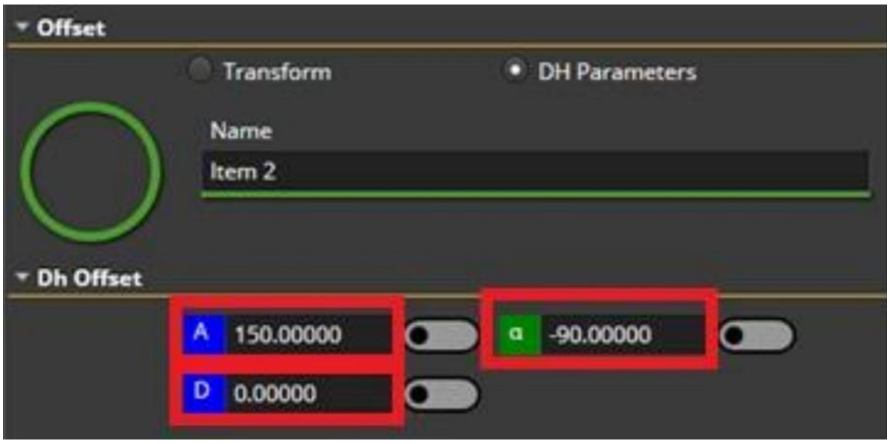
● Rotational ○ Linear ○ No Joint

Name: Link 1

Rotational Joint

- Is External
- Zero Offset: 0.00000 [°]
- Gear Ratio: 1.00000
- Backlash: 0.00000 [°]

while **R** (**A** in HxGN Robotic Automation), **α** and **D** are configured under **Dh Offset** properties:



Offset

○ Transform ● DH Parameters

Name: Item 2

Dh Offset

- A: 150.00000
- α: -90.00000
- D: 0.00000

5.2 Cell modelling

Robotic Automation application has the capability to model a cell layout which includes the following:

- **Machines**: such as robots, robots on a slider (robot extension), turn tables – part holders.
- **Tools**: predefined tools such Leica T-Scan or AS1 and reflectors.
- **Sensors**: predefined sensors such as Leica AT960.
- **Fixtures.**
- Any static object.

Prior to building the model you will need to prepare the following:

- CAD file (format such as STP, JT, etc.) for any object.
- Existing library should have machine model names created according to chapter 5.1.
- Tools and sensor model names used.
- Cell dimensions or drawings.

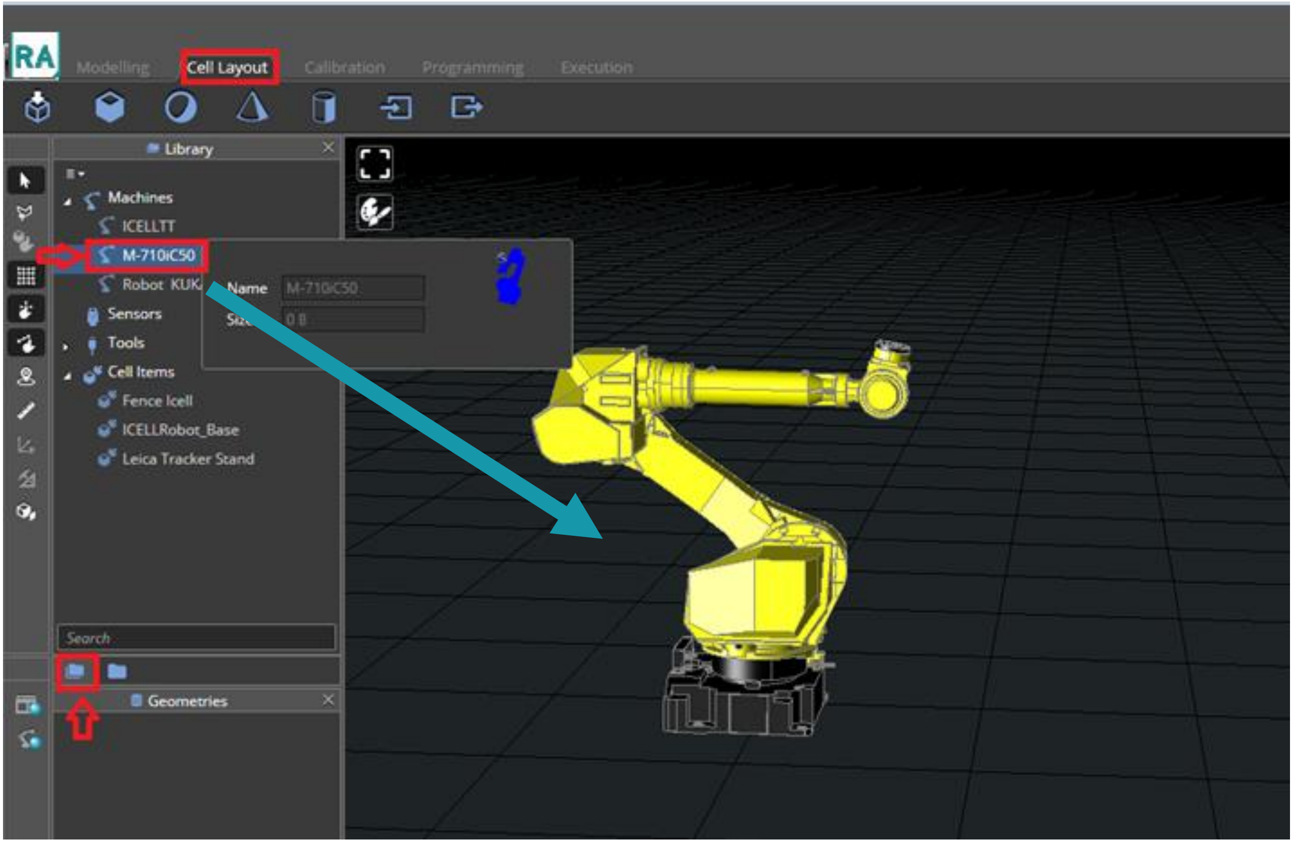
5.2.1 Adding items to the cell

When starting Robotic Automation, and creating a New Project, an empty cell layout (named Untitled) is opened in the 3D Viewer. In the Cell Layout the user can add items to set up a Cell either by importing items from the library or by importing them as CAD models. The two modes are explained in the following sections.

5.2.1.1 Importing items from the library

Cell Items from the Library as: Robots, Turn Tables, Fences, Cabinets, Robot, Controllers can be added to the cell from the library if they were previously created and exported.

On the Items Tree click on the **Library** Icon and expand the Machines Folder. E.g., if you want to use the FANUC M-710iC50, select it and drag it and drop it in the 3D Window.



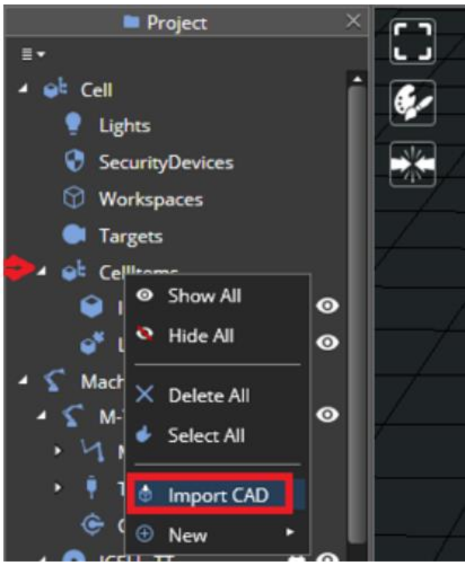
You can continue to add other items from the **Machines** Library, or static object from **Cell Items** library in the same way.

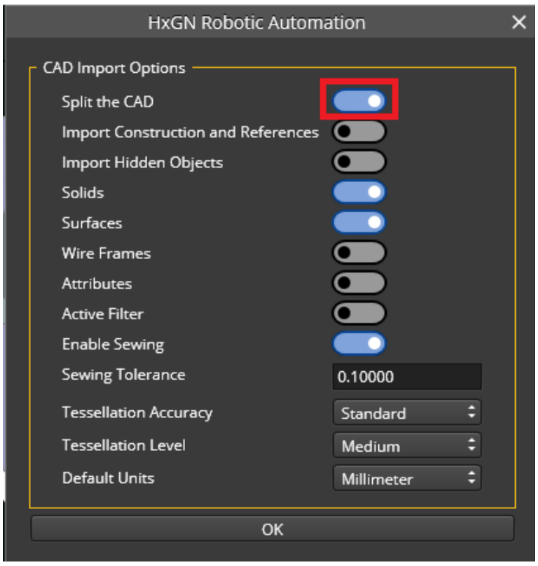
5.2.1.2 Importing items as CAD models

User can add Items to a Cell by importing CAD Models and editing their properties, right-click on **Cell Items** and select **Import CAD** and browse the CAD file name needed.

The **CAD Import Options** dialog will pop up. The **Split the CAD** option is disabled by default and can be enabled in case it is needed.

After clicking **OK**, you will have the cell with its components positioned according to the CAD coordinate system.





HxGN Robotic Automation

CAD Import Options

Split the CAD	●
Import Construction and References	○
Import Hidden Objects	○
Solids	●
Surfaces	●
Wire Frames	○
Attributes	○
Active Filter	○
Enable Sewing	●
Sewing Tolerance	0.10000
Tessellation Accuracy	Standard
Tessellation Level	Medium
Default Units	Millimeter

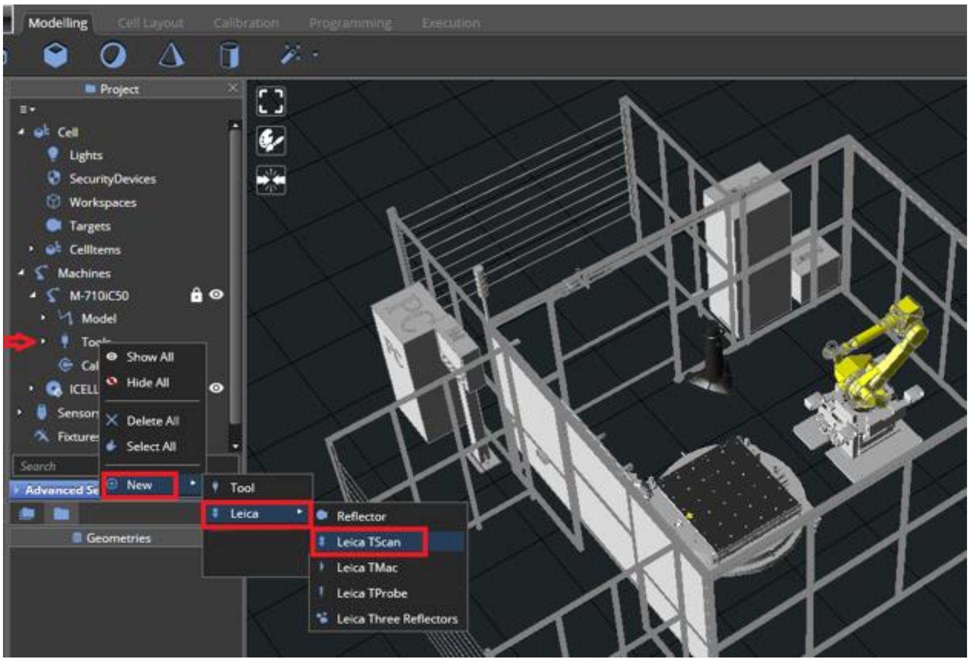
OK

5.2.1.3 Loading and attaching a Tool to the Robot

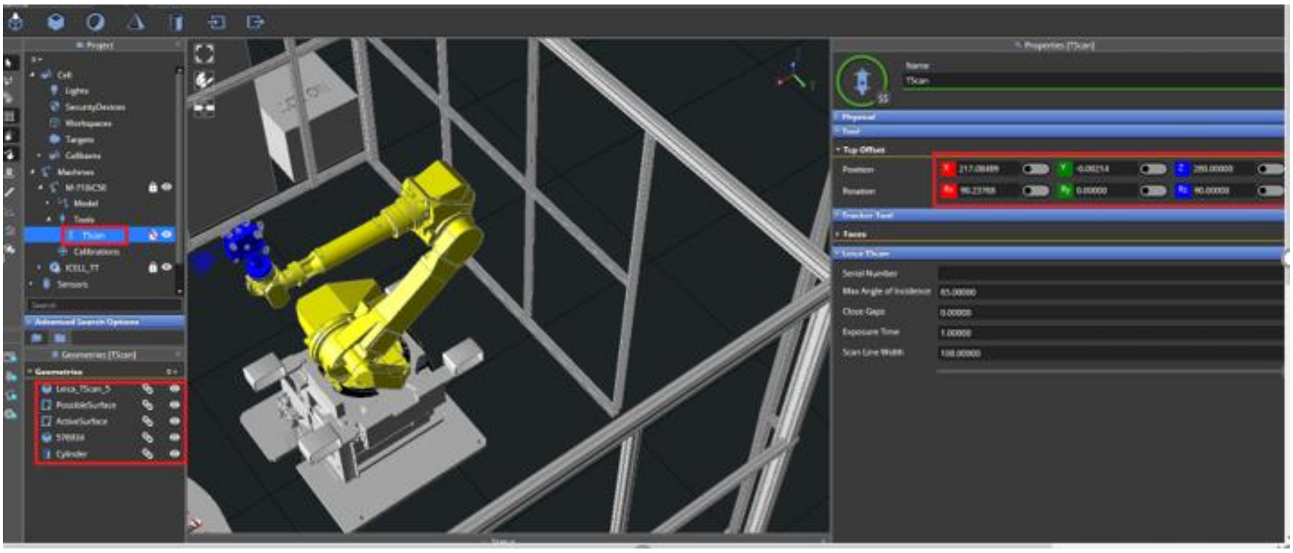
Once the Robot is added to the layout, a Tool to a Robot can be added as an existing Tool from the Library and Editing its position and activating it.

Once the project is loaded and Robot is displayed in the 3D Window:

1) Expand the **Machines** Tree.
2) Select the robot (e.g., M-710iC50) and expand it.
3) Right-click on the **Tools** section and select the tool, e.g., a Leica T-Scan 5 scanner (*New* | *Leica* | *Leica TScan*).



The **Tscan** tool will be added to the robot in the 3D Viewer. You can use the align tool and position the Tscan correctly on Robot J6 and add any Adaptor CADs as needed.



Name: TScan

Position: X 217.08499 Y -0.00214 Z 280.00000

Rotation: Rx 90.23768 Ry 0.00000 Rz 90.00000

Serial Number:

Max Angle of Incidence: 65.00000

Close Gaps: 0.00000

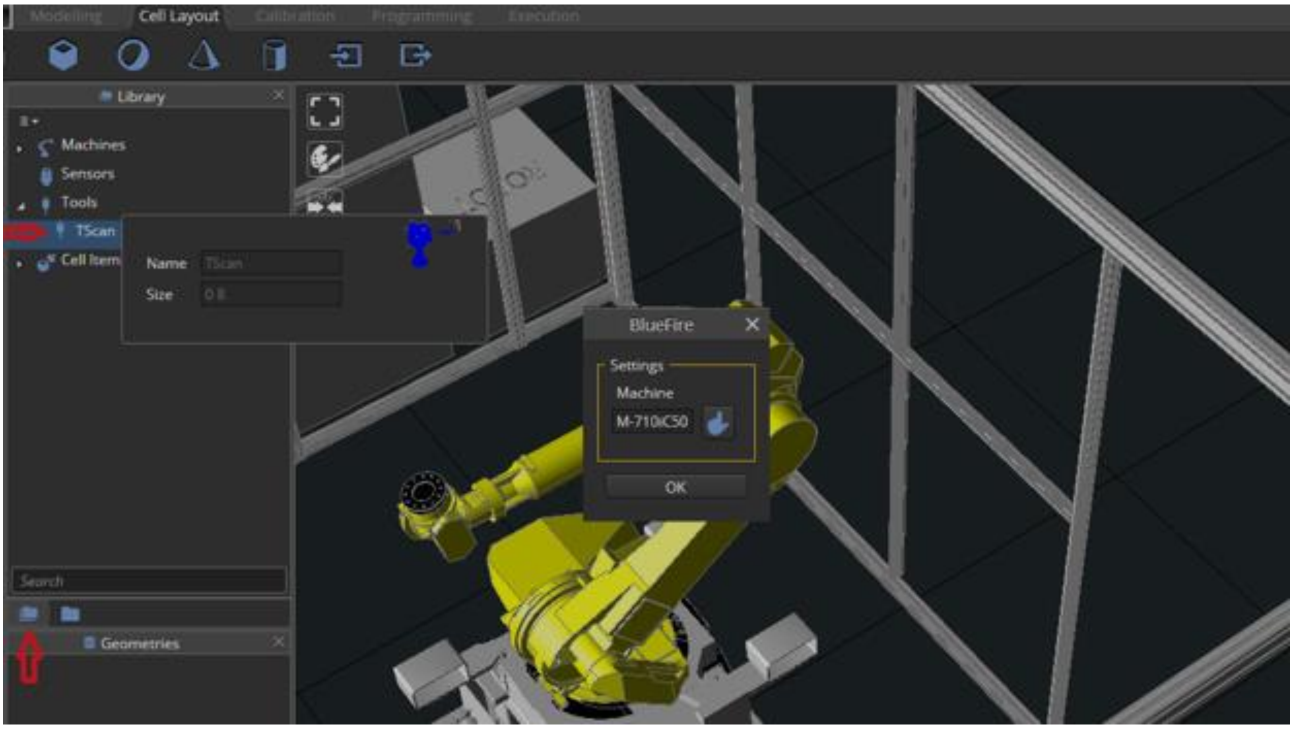
Exposure Time: 1.00000

Scan Line Width: 100.00000

Alternatively, it is possible to add a tool from the library.

The same steps followed to add other items from the library can be followed, see chapter 5.2.1.1. The only difference is that the tool will be found under **Tools**.

After dragging and dropping it into the 3D Window, a pop-up window will appear to select the Robot to which the tool will be attached where it is possible to verify that the Robot model type assigned is correct or select the right one by clicking on the hand icon and pressing **OK**.



Name: TScan

Size: 0 B

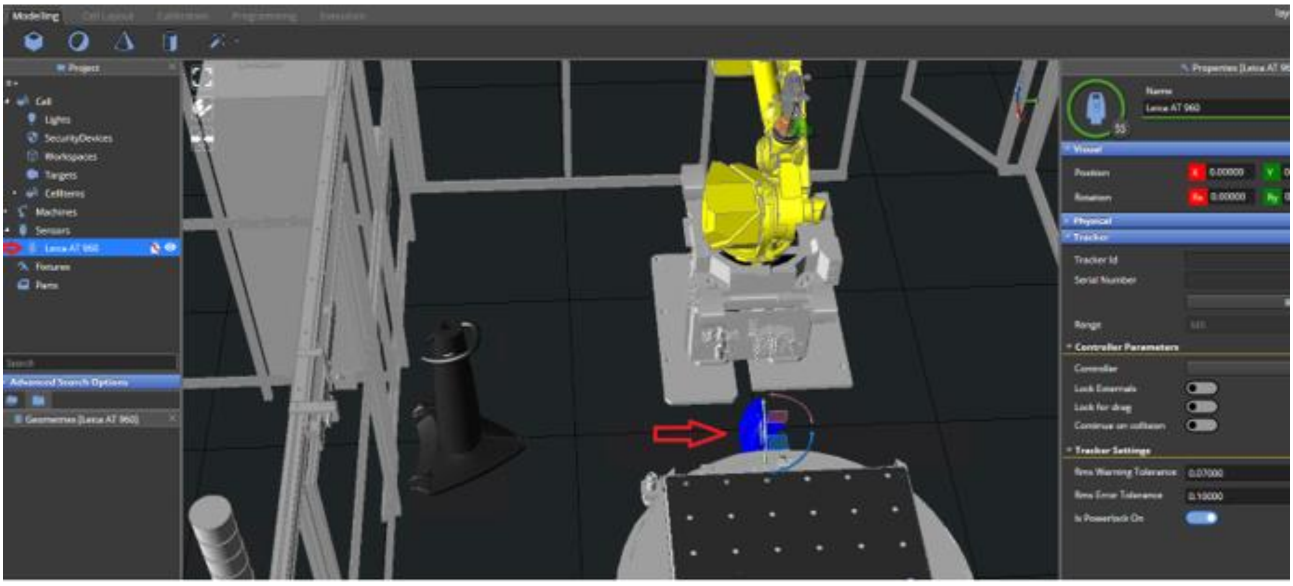
BlueFire

Settings

Machine: M-710iC50

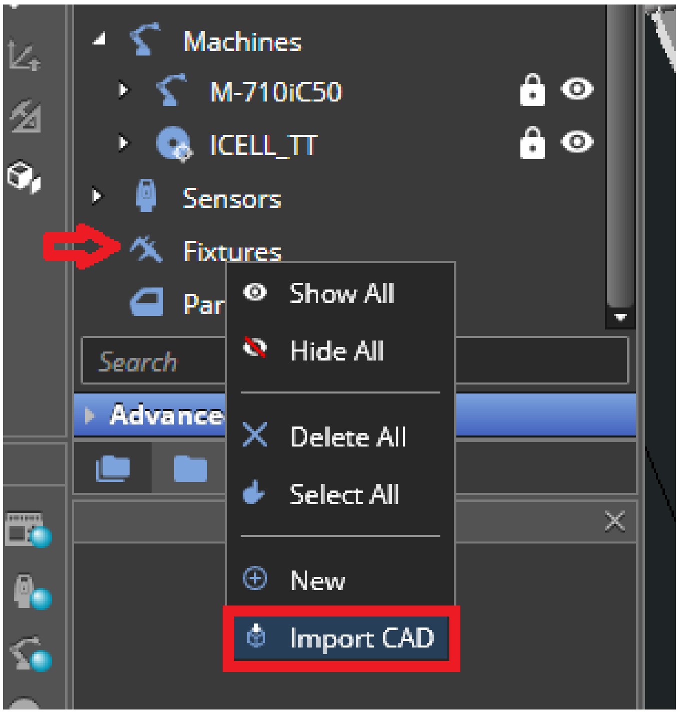
5.2.1.4 Adding a Sensor to the Cell

Right-click on the **Sensors** icon ( ) and select *New* | *Tracker* | *Leica AT960*.
The Leica AT960 will be added in its default position as shown on its properties.
The alignment tool (see chapter 5.2.3.1) can be used to position it as it is in real on top of the Leica Tracker stand under **Cell Items**.

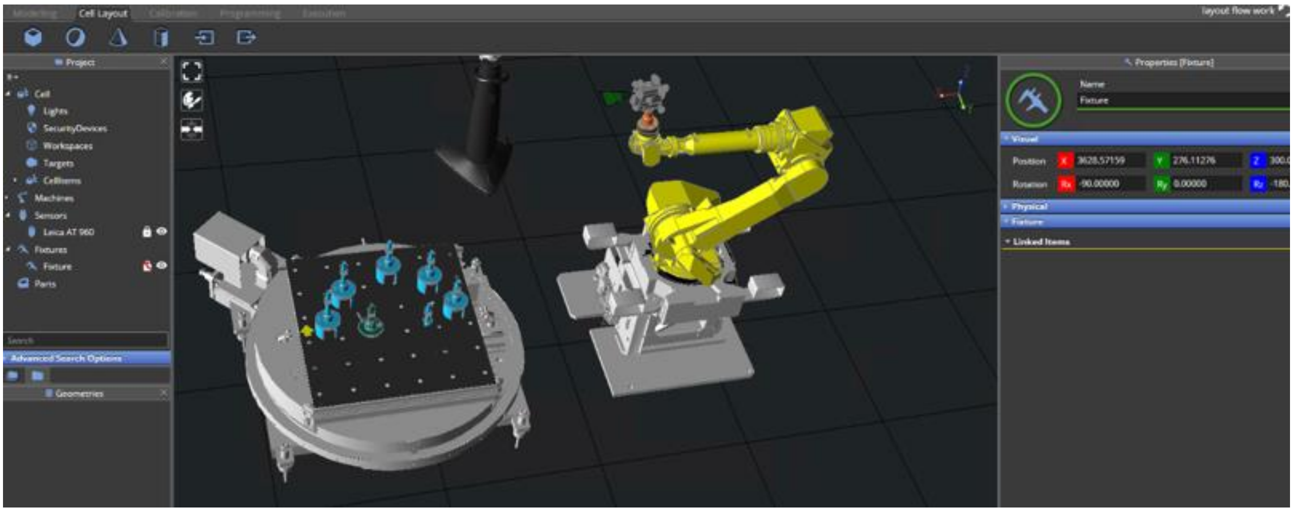


5.2.1.5 Adding a Fixture to the Cell

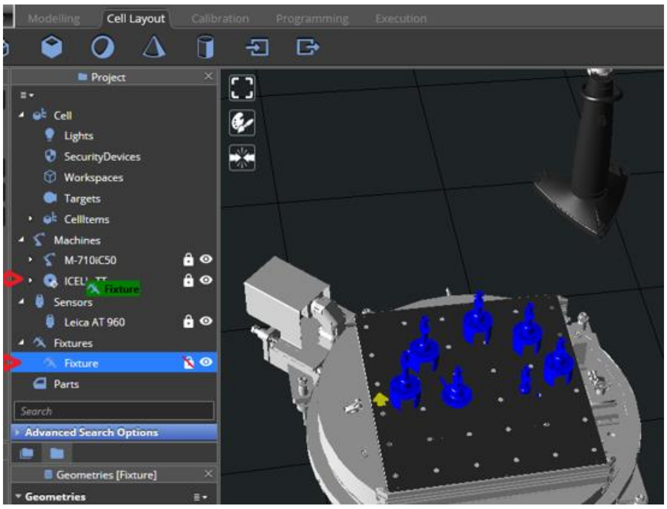
Right-click on **Fixtures** and **Import CAD**. Alternatively, you can use **Import CAD** icon ( ) in Cell Layout tab and once the CAD appears under cell item, you can assign graphic to it.

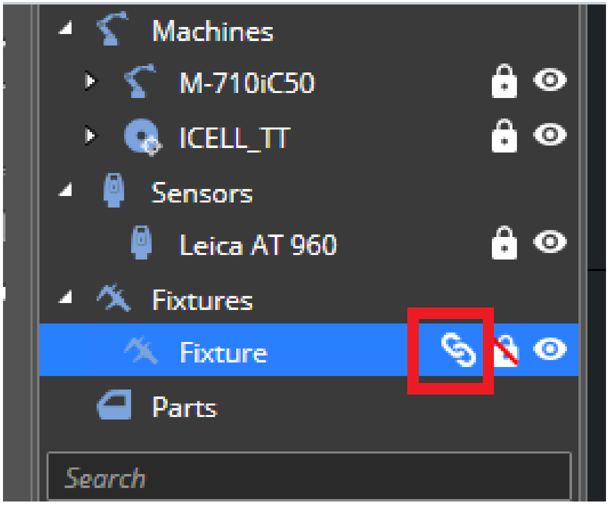


Align the fixture using the alignment tool (see chapter 5.2.3.1) to position it on the Turn table.



Link the Fixture to the Turn table by selecting the fixture with the mouse and dragging it under the Turn Table until it will turn to green, then drop it. The linked icon (🔗) will appear next to the fixture, meaning that it is linked.



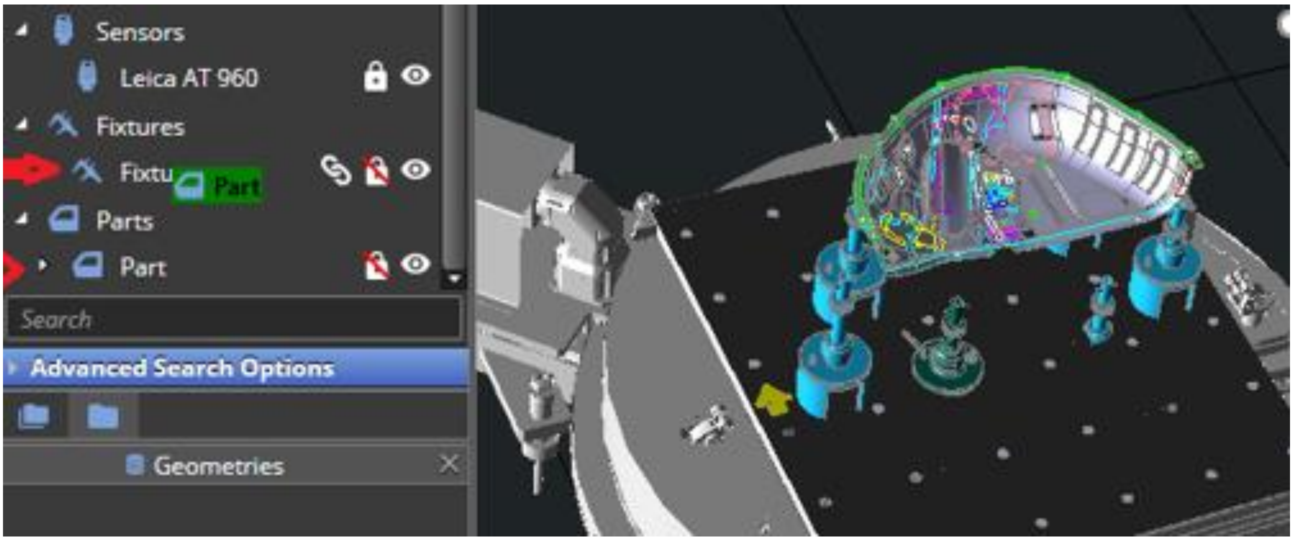


5.2.1.6 Adding a Part to the Cell

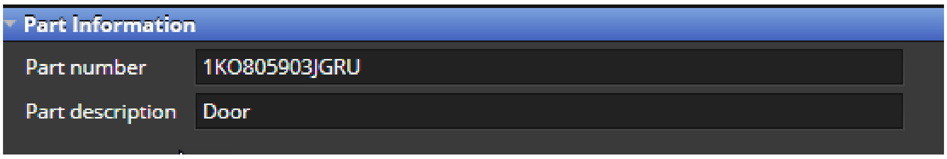
It is recommended to import the part from the metrology software.
Once the data from the metrology software project is transferred to the HxGN Robotic Automation project, the part's CAD will be added to a new part element in the tree view.

In alternative, the part can be directly imported in HxGN Robotic Automation by right-clicking on **Parts** and **Import CAD** and use the alignment tool (see chapter 5.2.3.1) and place it on the fixture.

Link the Part to the Fixture by selecting the part with the mouse and dragging it under the Fixture until it will turn to green, then drop it. The linked icon (🔗) will appear next to the fixture, meaning that it is linked.



Within the part properties, there is a section to add the part number and optionally a description. The part number is mandatory and will be requested upon exporting to Execution Manager in case it has not been added here.

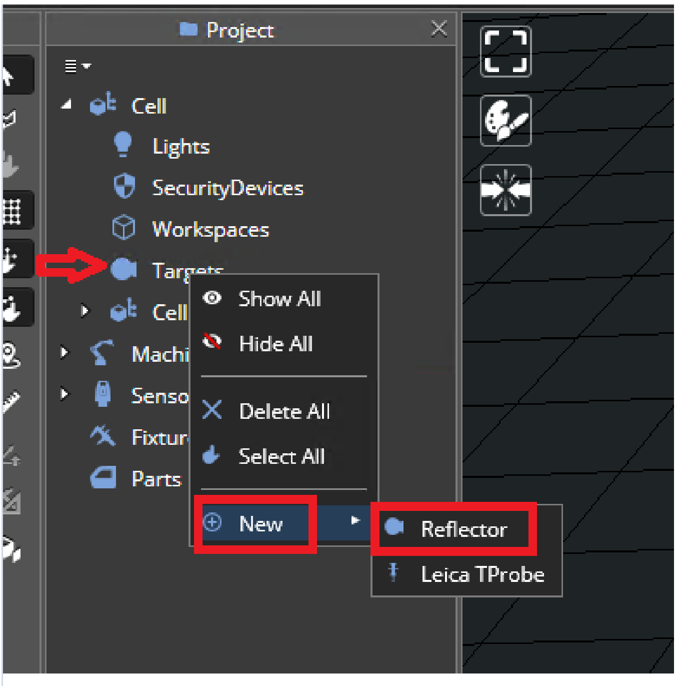


Part Information

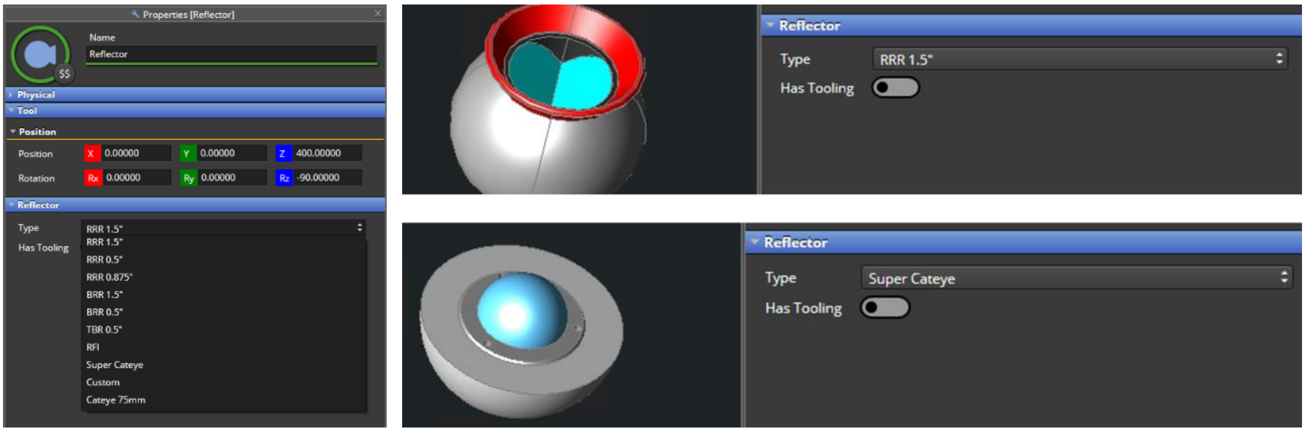
Part number	1KO805903JGRU
Part description	Door

5.2.1.7 Adding Targets to the Cell

Under **Cell** after expanding, right-click on *Targets* | *New* | *Reflector.* Reflectors are used for Calibration and Turn table compensation.



Add as many reflectors as the real cell has, double click on them and set their properties, e.g., by setting their position on the Turn table. Their type can be selected from the **Type** drop-down list.



Properties [Reflector]

Name: Reflector

Physical

Tool

Position

Position	X 0.00000	Y 0.00000	Z 400.00000
Rotation	Rx 0.00000	Ry 0.00000	Rz -90.00000

Reflector

Type: RRR 1.5"

Has Tooling

RRR 1.5"
RRR 0.5"
RRR 0.875"
BRR 1.5"
BRR 0.5"
TBR 0.5"
RFI
Super Cateye
Custom
Cateye 75mm

Reflector

Type: RRR 1.5"

Has Tooling

Reflector

Type: Super Cateye

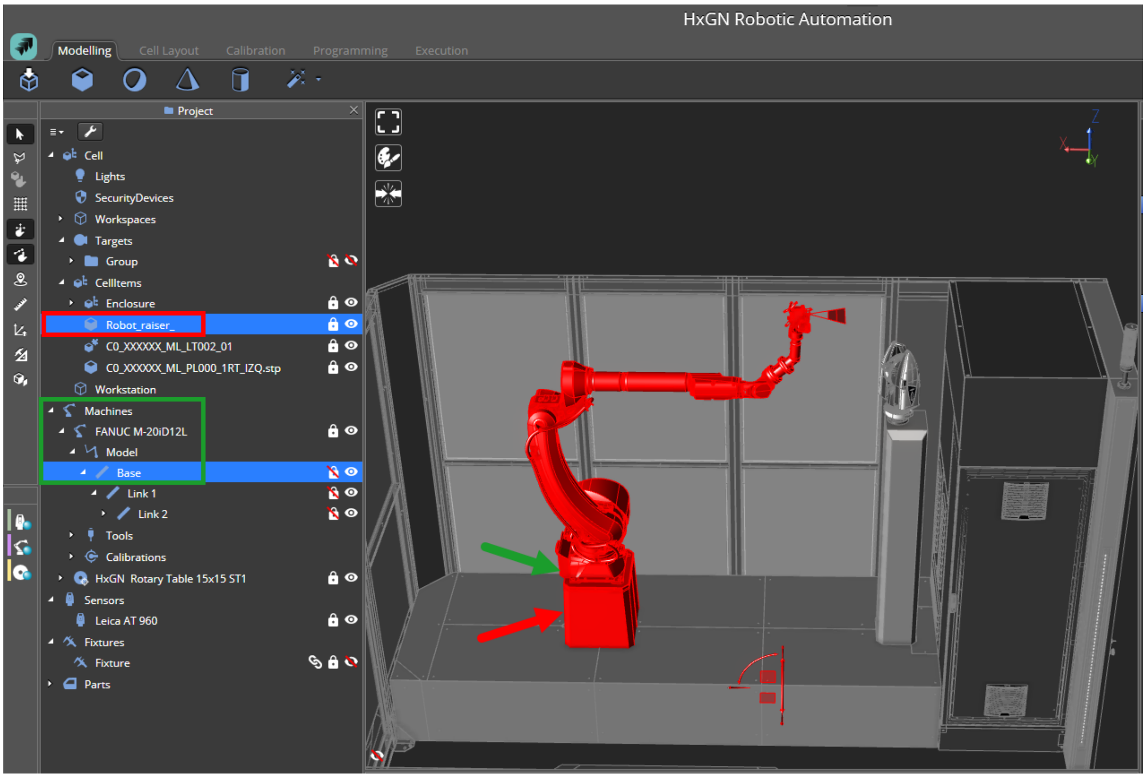
Has Tooling

5.2.2 Defining Collisions rules among cell items

HxGN Robotic Automation provides a collision engine that is checking if cell items are colliding with each other in the 3D view. Collision rules are established by default among all cell items but can be freely defined by the user.

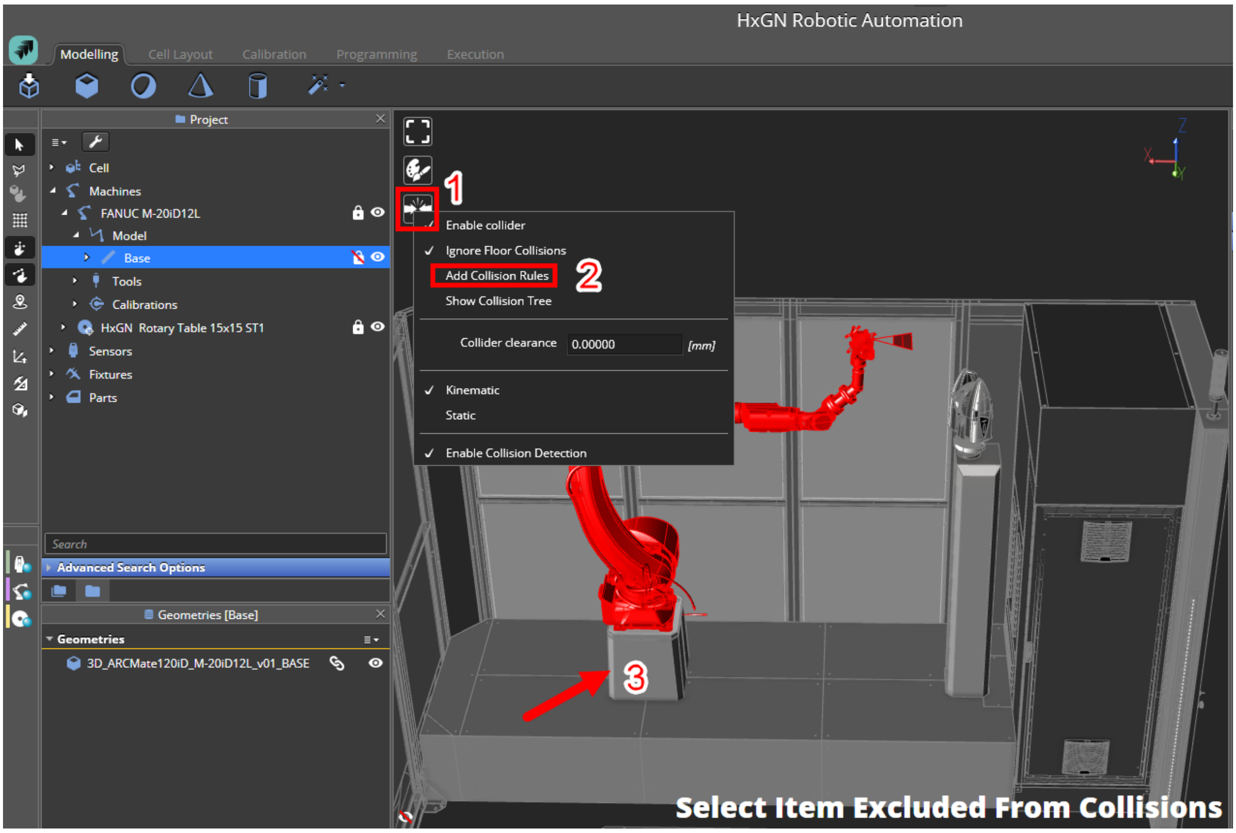
To create a collision rules among two cell items, the user needs to:

1) select the cell item where he needs to create a collision rule for, e.g. Robot **M-20iD12L** as shown below.
2) Select a part of it, e.g. its **Base** can be selected to create a collision rule between it and the robot raiser (**Robot_raiser_** in the example).



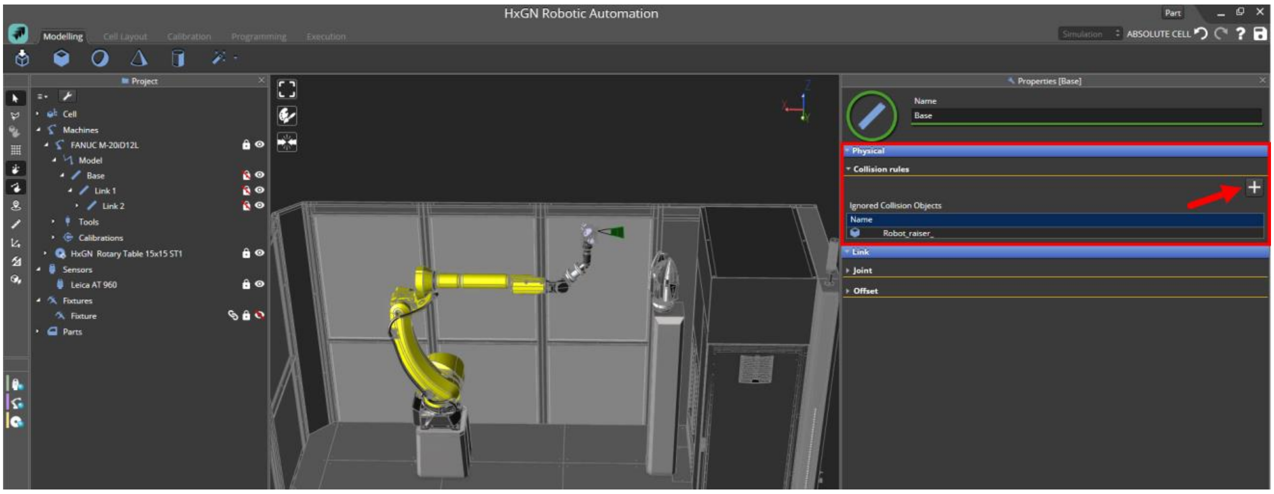
Then the user can either:

1) click on the **Collisions Overlay** Icon in the 3D View
2) select **Add Collisions Rules** from the Context Menu
3) in the 3D Viewer click on the CAD of the item to be excluded from the collision check for the Robot Base, in this example, the robot stand is excluded. Alternatively the user can select the same item in the project tree. In both cases, it is needed to press the [ENTER] key to confirm.



Select Item Excluded From Collisions

Collision rules can also be added inside the item properties by clicking the **+** button and following the same as described above.



The collision rule will appear within *Physical* | *Collision rules* | *Ignored Collision Objects.*

This means that the collision engine will not check for collisions for the items that are available within the list of **Ignored Collision Objects**.

5.2.3 Arranging items into the cell

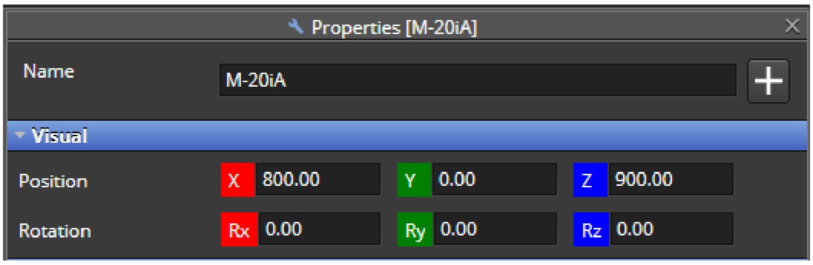
5.2.3.1 Alignment tool

Using the Alignment tool user can place/align items in the 3D Viewer relative between each other, based on alignment constraints as:

- Parallelism
- Concentricity & Coaxiality

Example – How to Align a Fanuc M-20iA Robot to a Robot Stand

On the Items Tree click on the Library Icon and expand the Machines Folder. Expand the FANUC Folder and select the M-20iA Robot and drag it and drop it in the 3D Window. Change its position to the initial coordinates:



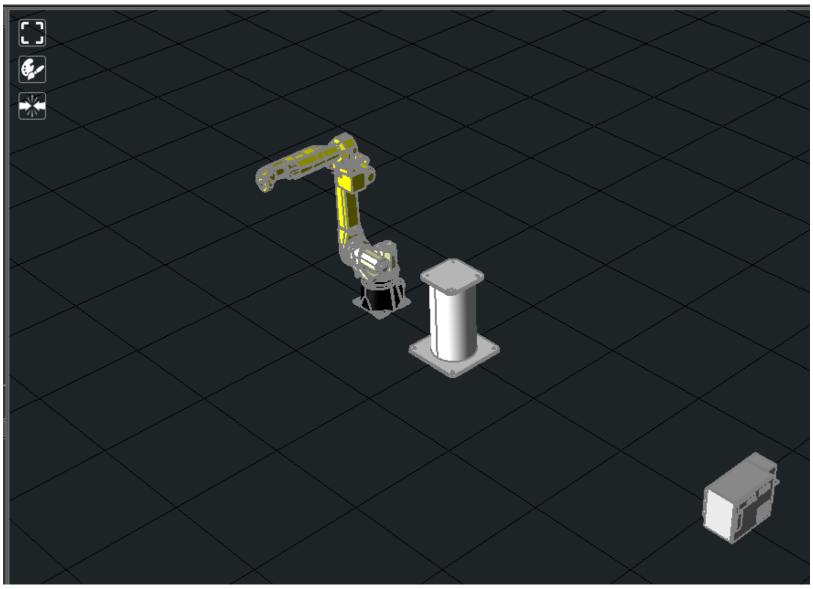
Properties [M-20iA]

Name: M-20iA

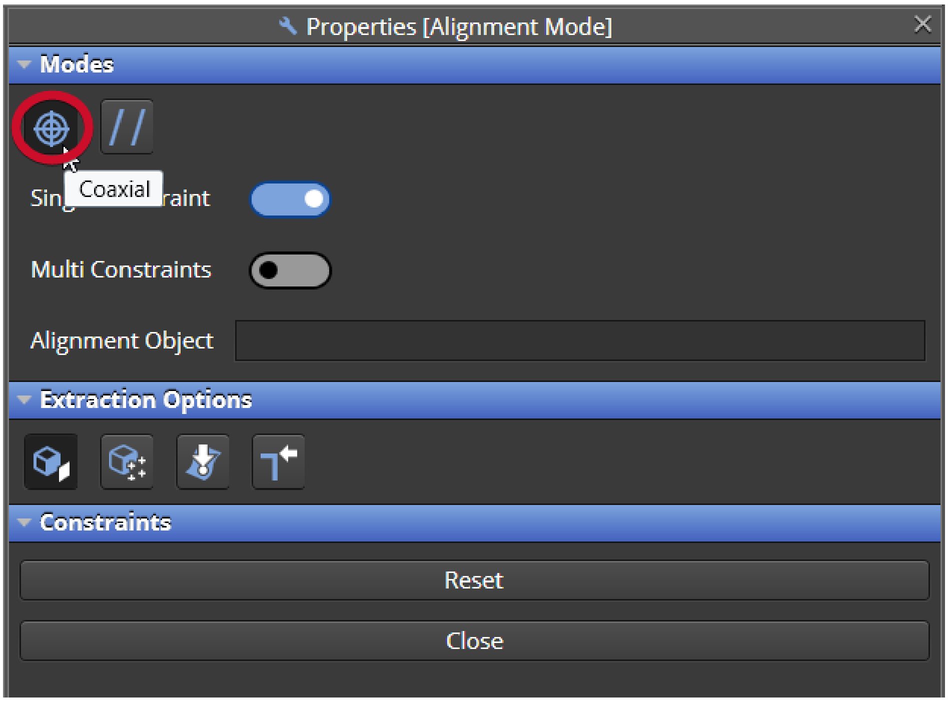
Visual

Position	X 800.00	Y 0.00	Z 900.00
Rotation	Rx 0.00	Ry 0.00	Rz 0.00

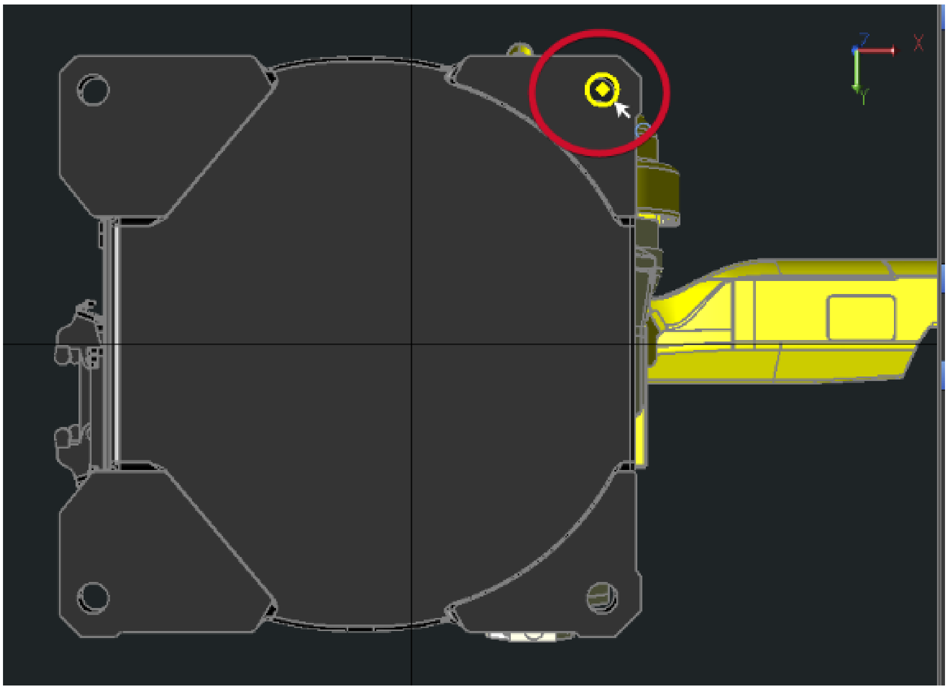
And the 3D Window should look like the image below:



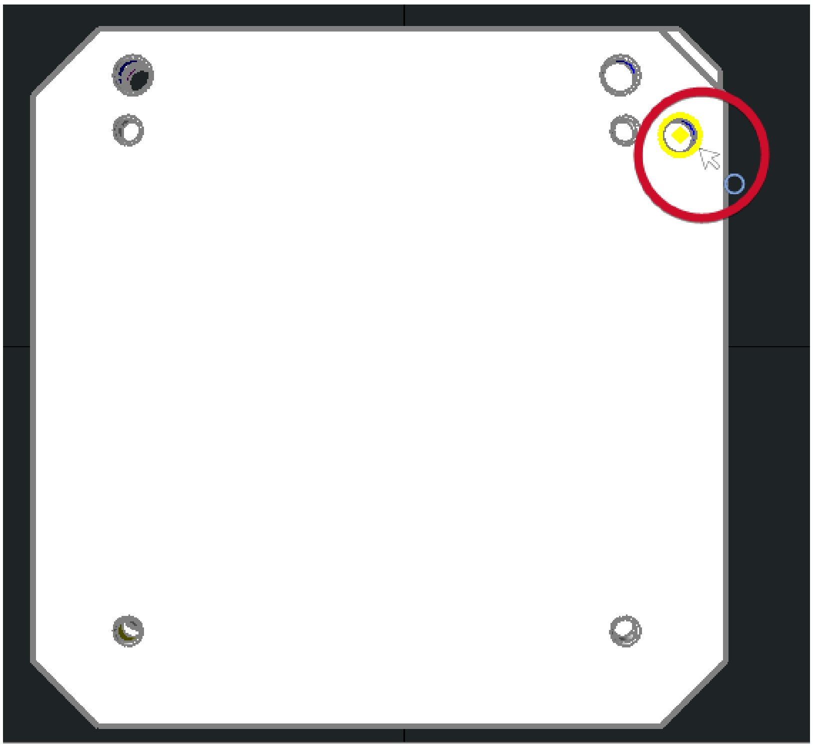
Next step is to align the robot to the top of the robot stand. To do this it is needed to activate the alignment mode by clicking on the Alignment Mode Icon in the Left Panel and generate a **Coaxial** constraint between the Robot Stand and the Robot base to place the Robot on top of the Stand within the Properties window.
The Robot is the item that will be moved to the Robot Stand (which is the item that is fixed).



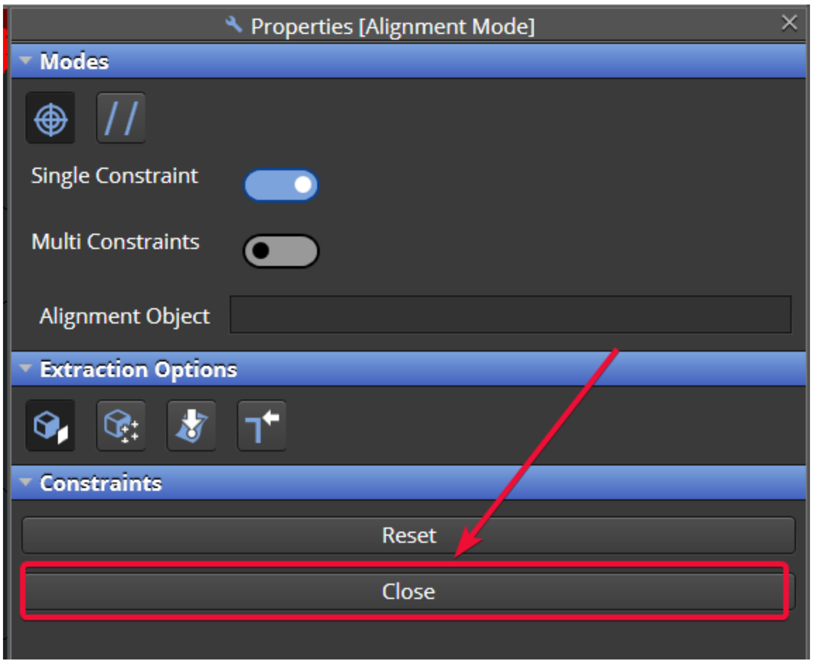
Rotate the 3D Window view mouse over the Robot Base bottom face and select the circle displayed on the image below:



And then rotate the 3D Window view, mouse over the Robot Stand top face and select the Circle displayed on the image below:



Once done we can close the alignment mode by clicking on **Close** button.



Properties [Alignment Mode]

Modes

Single Constraint ●

Multi Constraints ○

Alignment Object

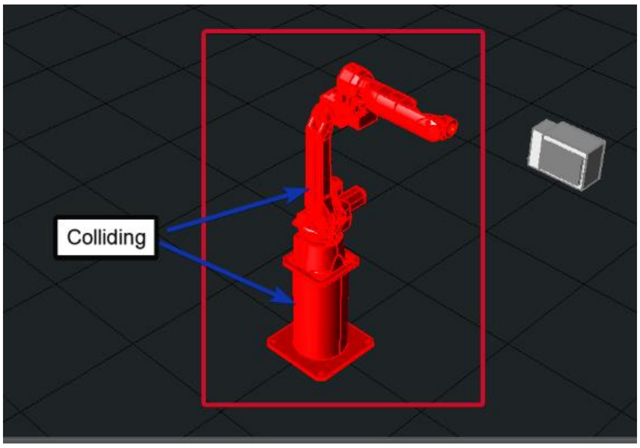
Extraction Options

Constraints

Reset

Close

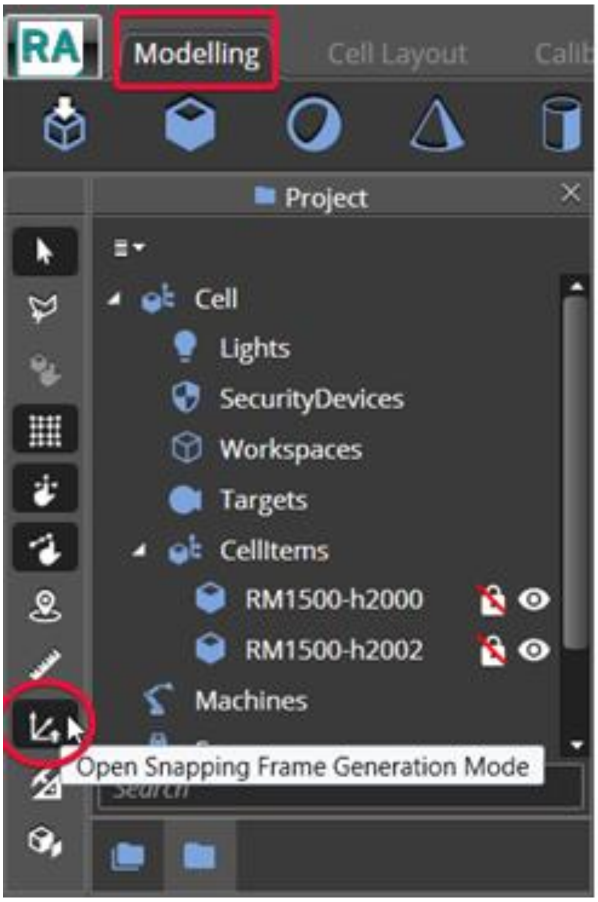
After clicking on the base of the alignment the robot will move on top of the robot stand and will show collision with it.



A new collision rule between the Robot Base and the Robot Stand should be created as explained in chapter 5.2.2 so these collisions are filtered.

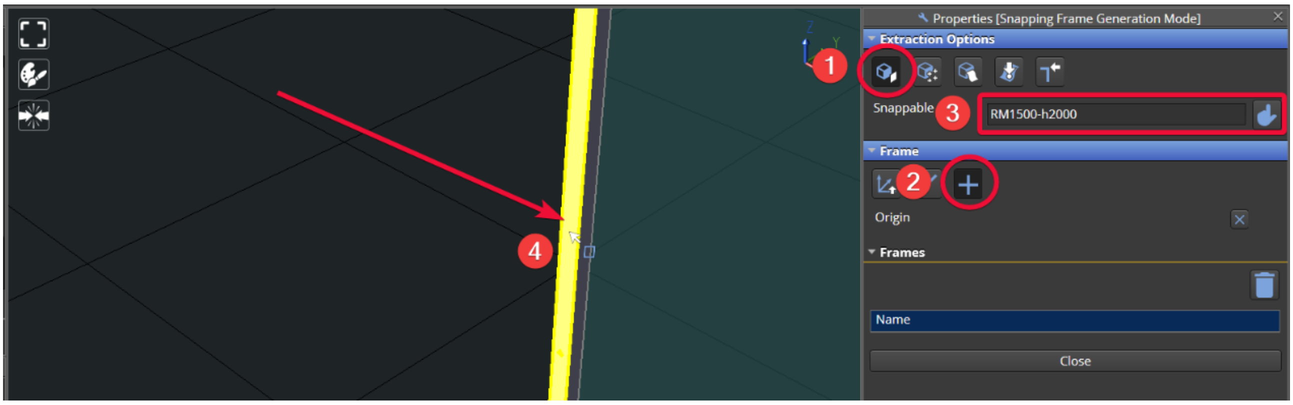
5.2.3.2 Snapping frames

A snapping frame is a point added to a cell item.
Once two objects with snapping frames get closer, a bubble appears and if we bring them close enough and stop dragging, the items will be aligned by the snapping frame.
The snapping frame generation mode can be opened in the **Modelling** tab.



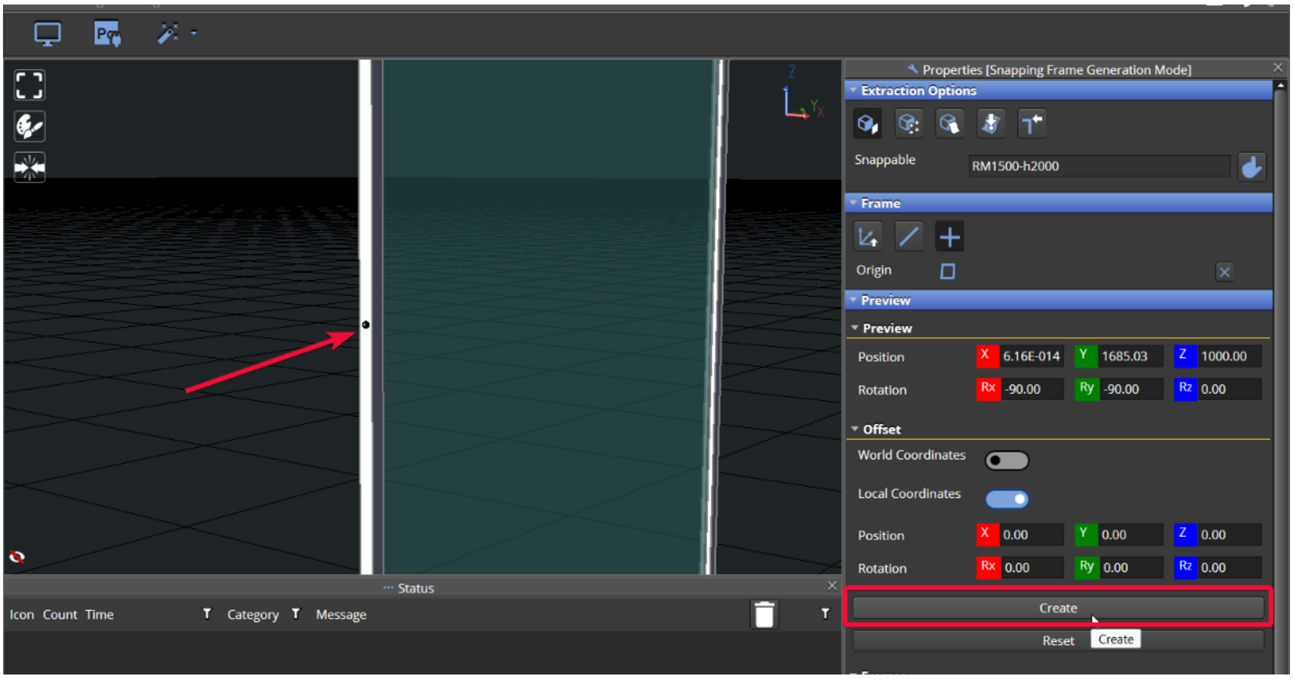
The following steps must be followed:

1) Select feature on extracting options.
2) Select point on frame.
3) Select the fence where you want the snapping frame to be generated.
4) Zoom in until you can select the side of the fence.



Snappable: RM1500-h2000
Origin

Once selected, a point will appear, and it is just needed to click on it and the frame will be created.

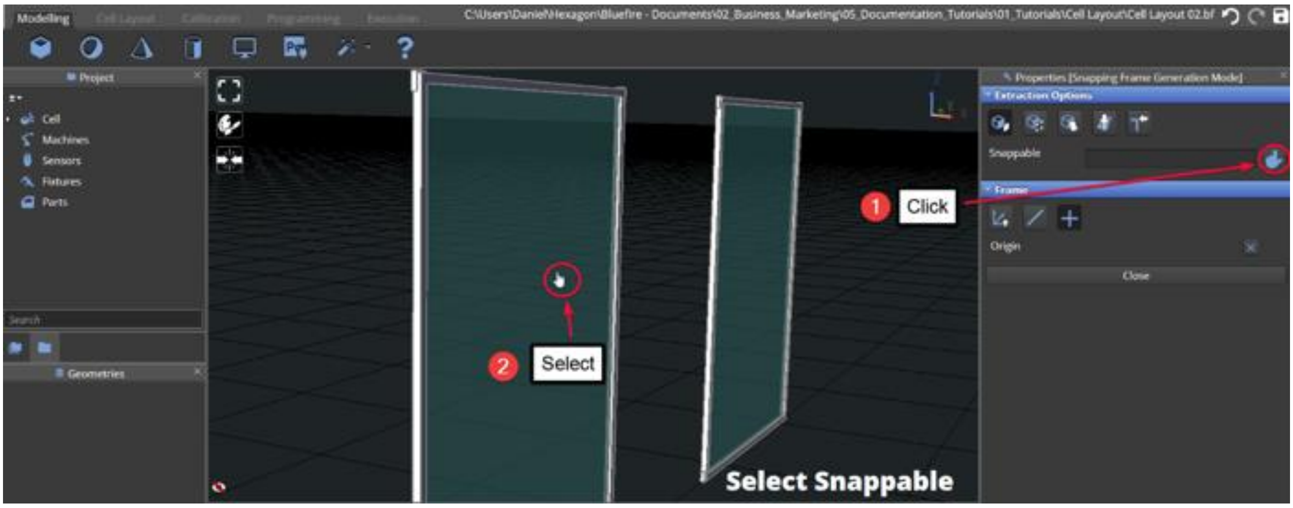


Snappable: RM1500-h2000
Origin

Preview
Position: X 6.16E-014, Y 1685.03, Z 1000.00
Rotation: Rx -90.00, Ry -90.00, Rz 0.00

Offset
World Coordinates: off
Local Coordinates: on
Position: X 0.00, Y 0.00, Z 0.00
Rotation: Rx 0.00, Ry 0.00, Rz 0.00

Then click on the other fence and create another frame on the opposite fence:

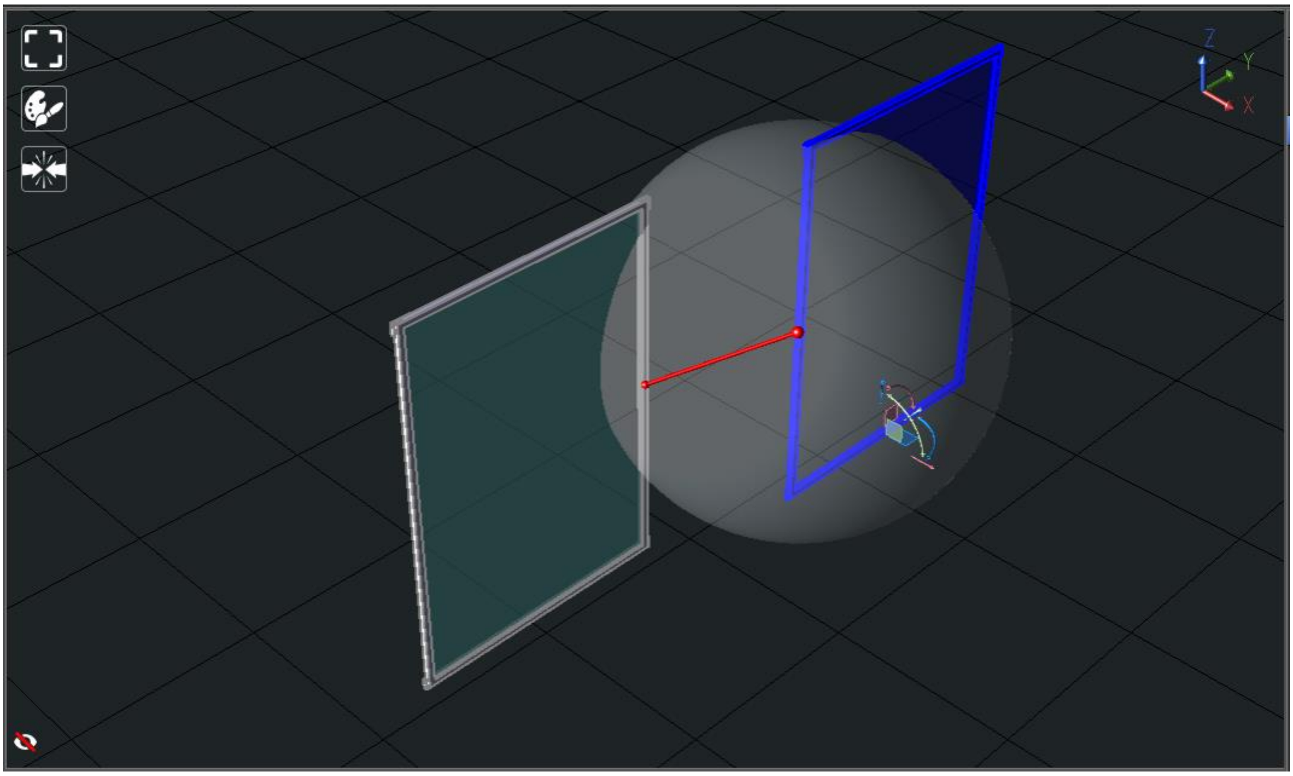


1 Click

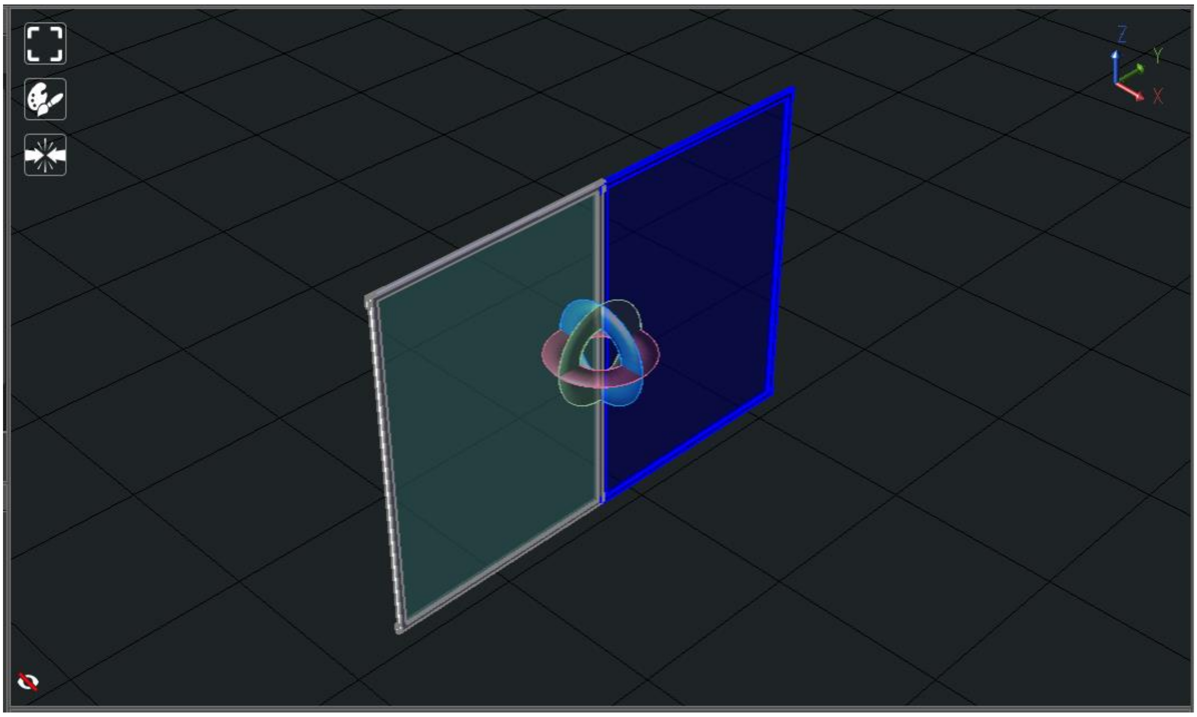
2 Select

Select Snappable

Then drag it near the other fence.



After releasing the fences will be automatically attached by the snapping frame point.



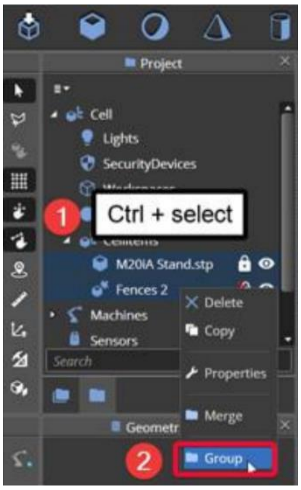
5.2.4 Grouping and Merging Cell Items

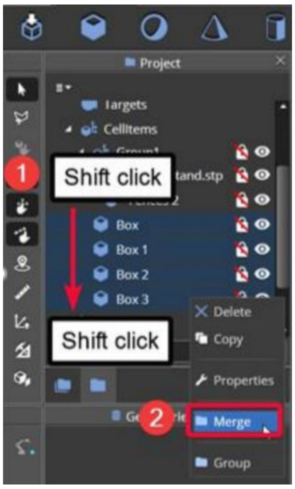
When Modelling a Machine or Building a Cell Layout there is often the need to **Group** Items to keep them under a same section or folder or **Merge** Items to make them a single assembly. After selecting the items and right-clicking on the selection, it is possible to either merge or group them by clicking the respective button.

5.2.4.1 Items selection

Items within the project tree can be selected in two ways:

1) Selecting them from the 3D Window by clicking and pressing the [CTRL] key for multiple choice.
2) Selecting them in the Project Tree using the range selection, clicking on the first element of the selection, and then pressing [SHIFT] key clicking on the last element of the selection.

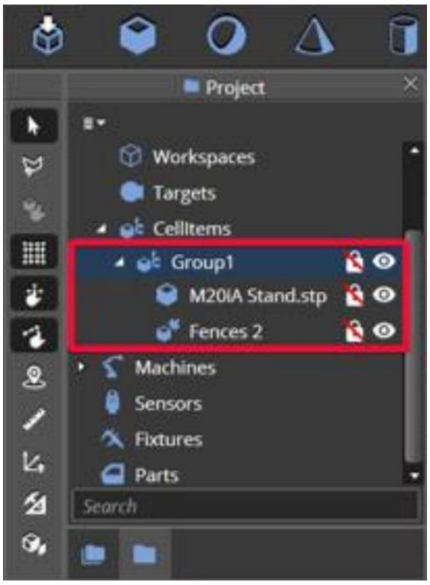




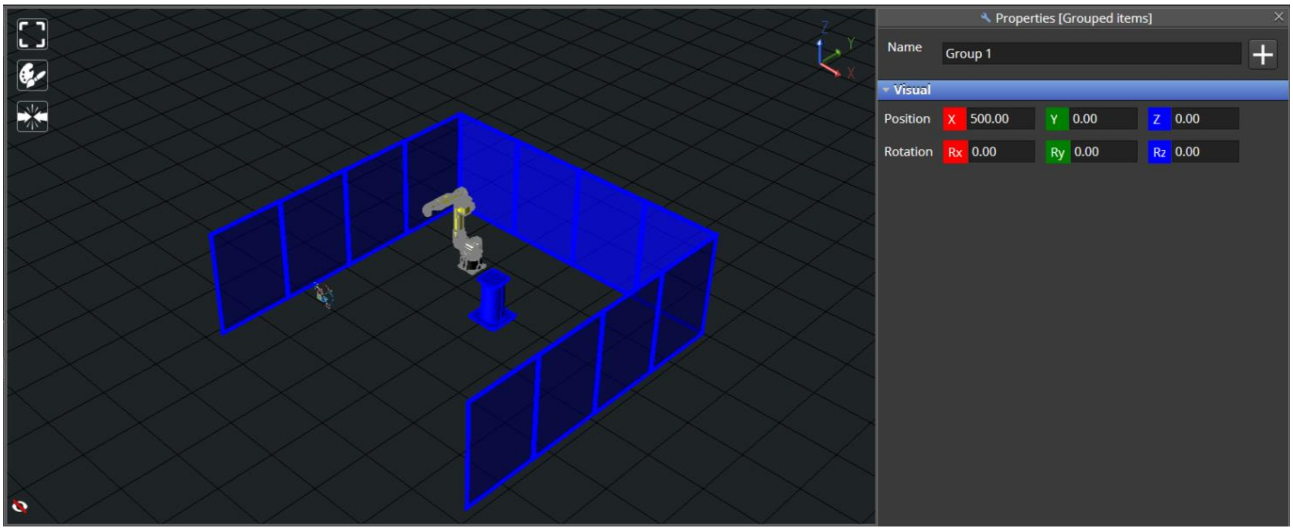
5.2.4.2 Grouping items

Grouped items properties can be edited individually or all together as a Group.
Consider for example the following project where two items are grouped under **Group1**:

- M20iA Stand.stp
- Fences 2



It is possible to move the entire Group1 by double-clicking on the group and changing the group position in the Properties, e.g., by moving the group along the X axis by 500 mm.

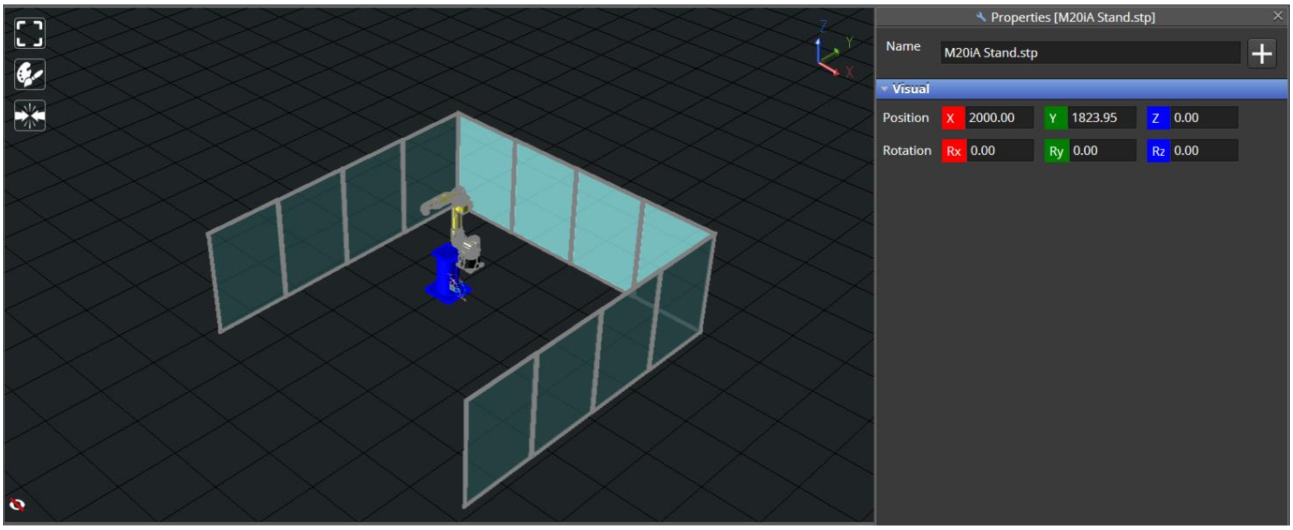

Properties [Grouped items]

Name: Group 1

Visual

Position	X	500.00	Y	0.00	Z	0.00
Rotation	Rx	0.00	Ry	0.00	Rz	0.00

Even though the Robot Stand is in the Group1 it can be edited individually without changing the rest of the Items in the Group1 position, e.g., by selecting the robot stand and moving it along the X axis by 2000 mm.



Properties [M20iA Stand.stp]

Name: M20iA Stand.stp

Visual

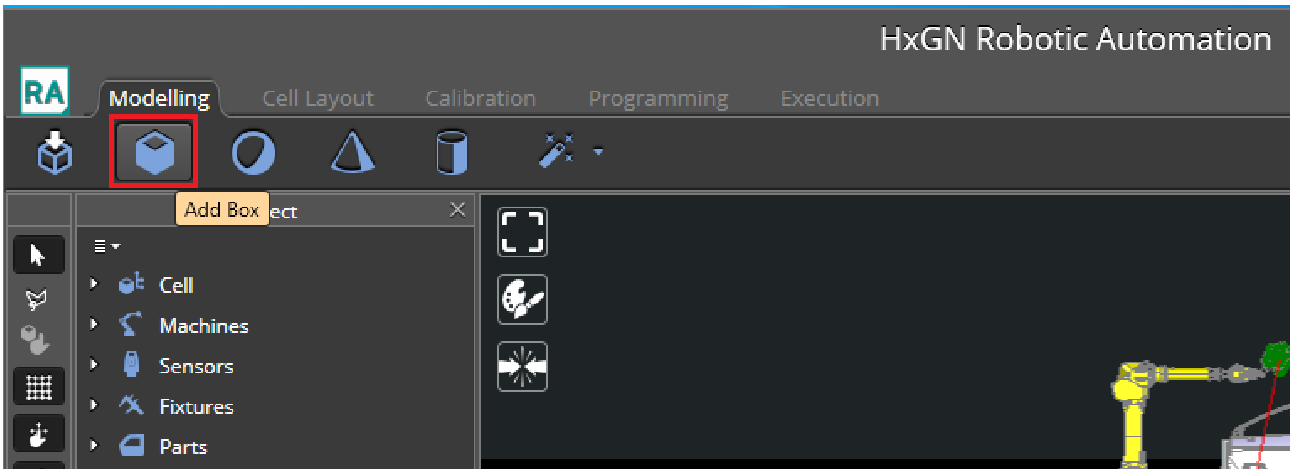
Position: X 2000.00, Y 1823.95, Z 0.00

Rotation: Rx 0.00, Ry 0.00, Rz 0.00

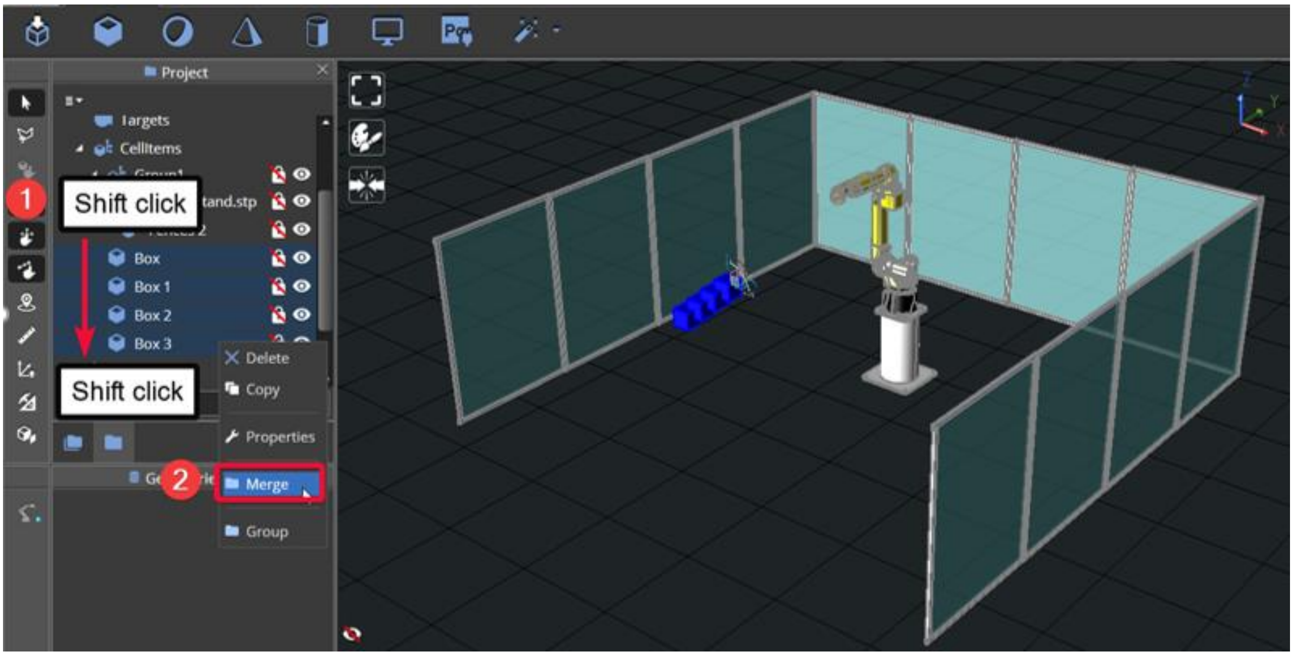
5.2.4.3 Merging items

Merged items will belong to the same assembly and cannot be moved individually.

To show the difference between Merging and Grouping it is possible to add four boxes to the Cell by clicking four times on the **Add Box** icon on the top toolbar.



Select the four boxes in the Items Tree using the range selection. Once all the boxes are selected, right-click and select **Merge** from the context Menu.



A New Merged Cell Item is created that has the 4 Boxes in a Single Item. None of the boxes can be translated individually.

6 Features

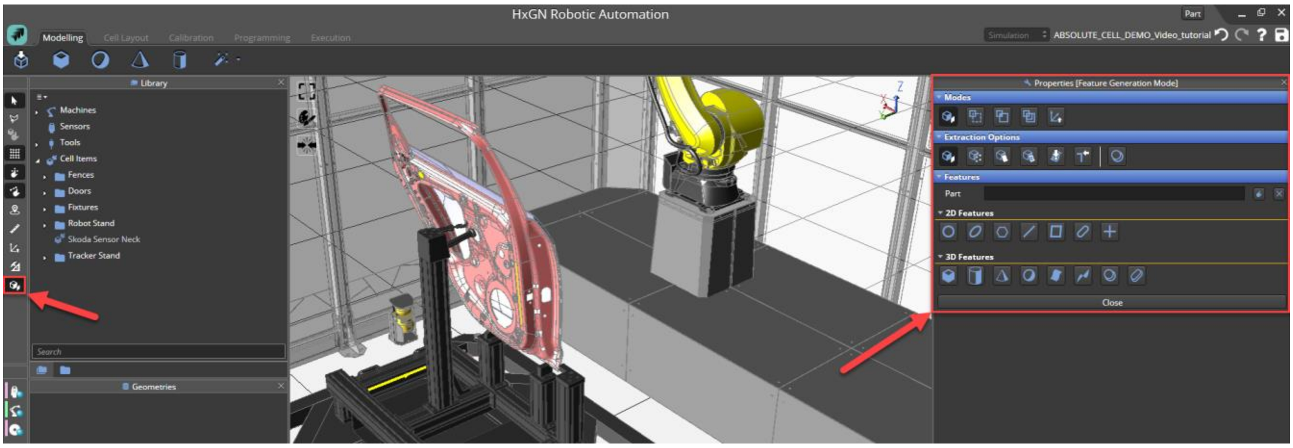
Features is a term used in metrology to refer to specific parts of a piece to be measured with a certain level of accuracy and are usually defined within an inspection plan and managed inside a metrology software. Alternatively, they can be generated inside HxGN Robotic Automation using the specific *Feature Generation Mode.*

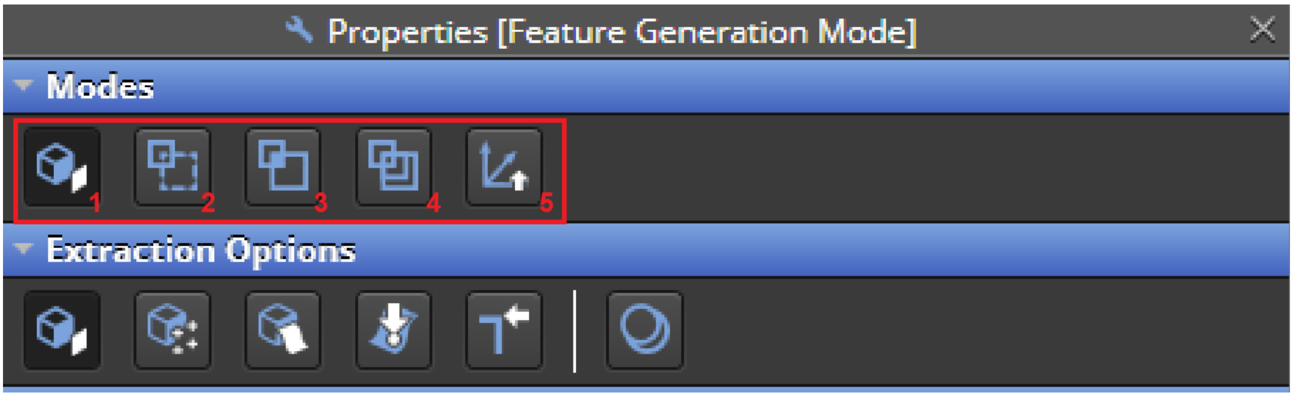
6.1 Import features from Metrology software

Features can be imported in HxGN Robotic Automation from metrology software with a specific workflow that depends on the metrology software used. Please refer to chapter 0 for further details.

6.2 Feature Generation Mode

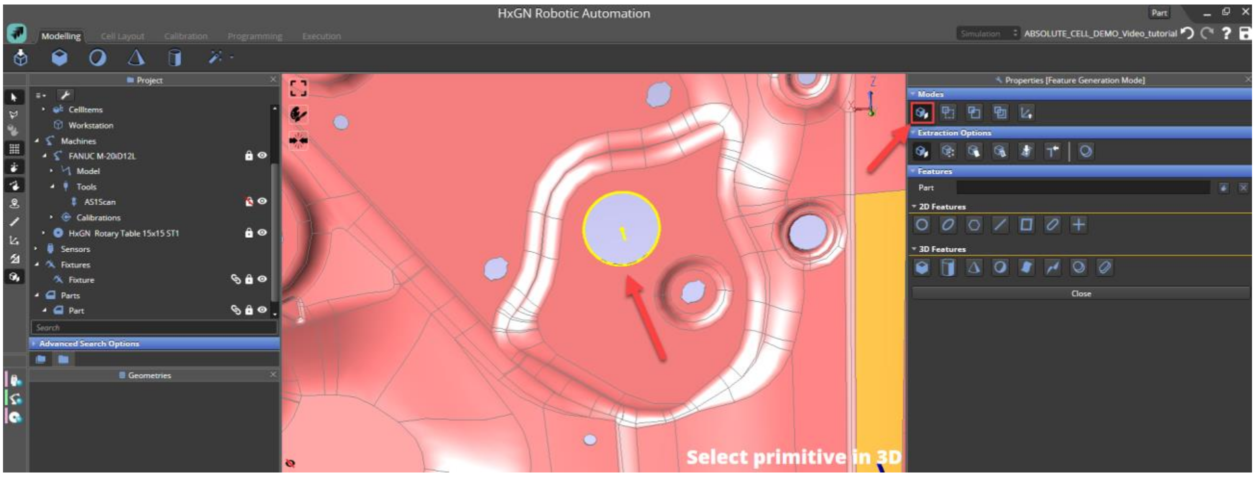
The user can generate features from HxGN Robotic Automation using the feature generation mode. After clicking on the Feature Generation Mode button on the special modes' toolbar, in the Properties window it will be possible to select among different options defining how to create features.

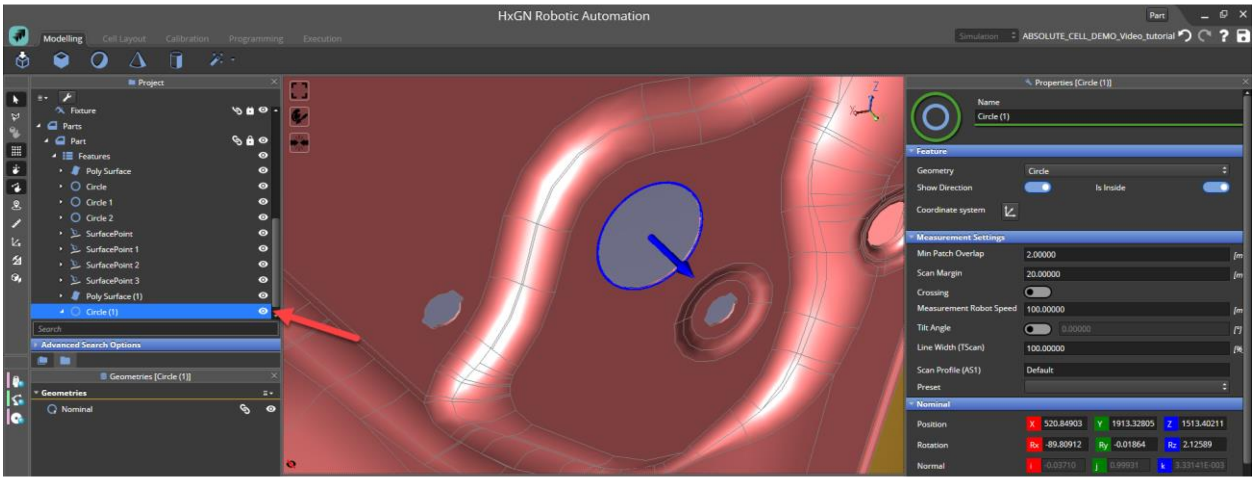




The extraction modes are:

1) **Extract from CAD**: the user needs to select the features by clicking on the CAD of the part in the 3D View as shown in the image below. After selecting the feature (e.g. a circle), the feature will be added to the Project Tree.





Name: Circle (1)

Feature

- Geometry: Circle
- Show Direction: on
- Is Inside: on
- Coordinate system

Measurement Settings

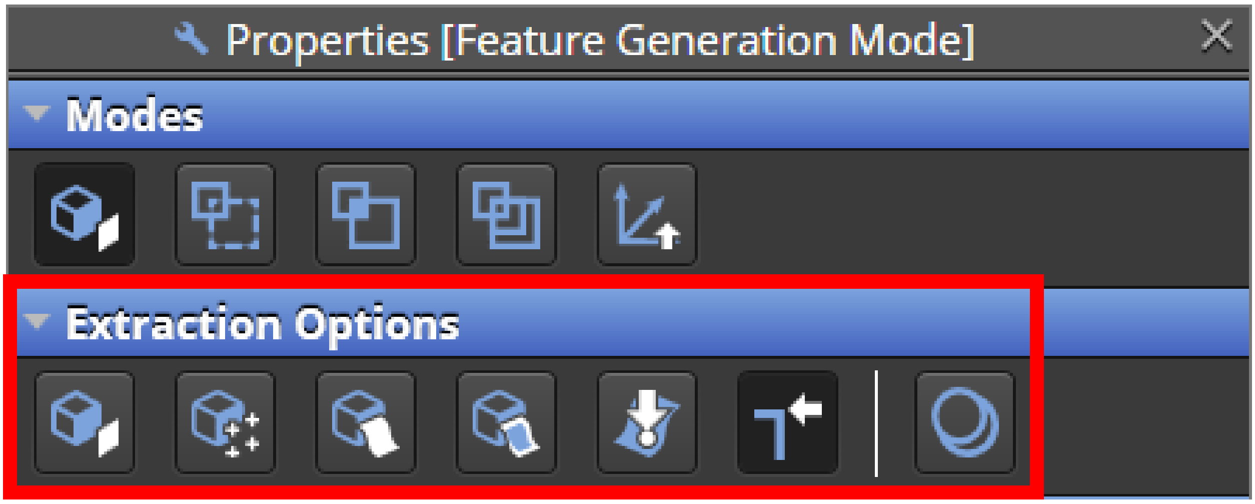
- Min Patch Overlap: 2.00000
- Scan Margin: 20.00000
- Crossing: off
- Measurement Robot Speed: 100.00000
- Tilt Angle: off, 0.00000 [°]
- Line Width (TScan): 100.00000
- Scan Profile (AS1): Default
- Preset

Nominal

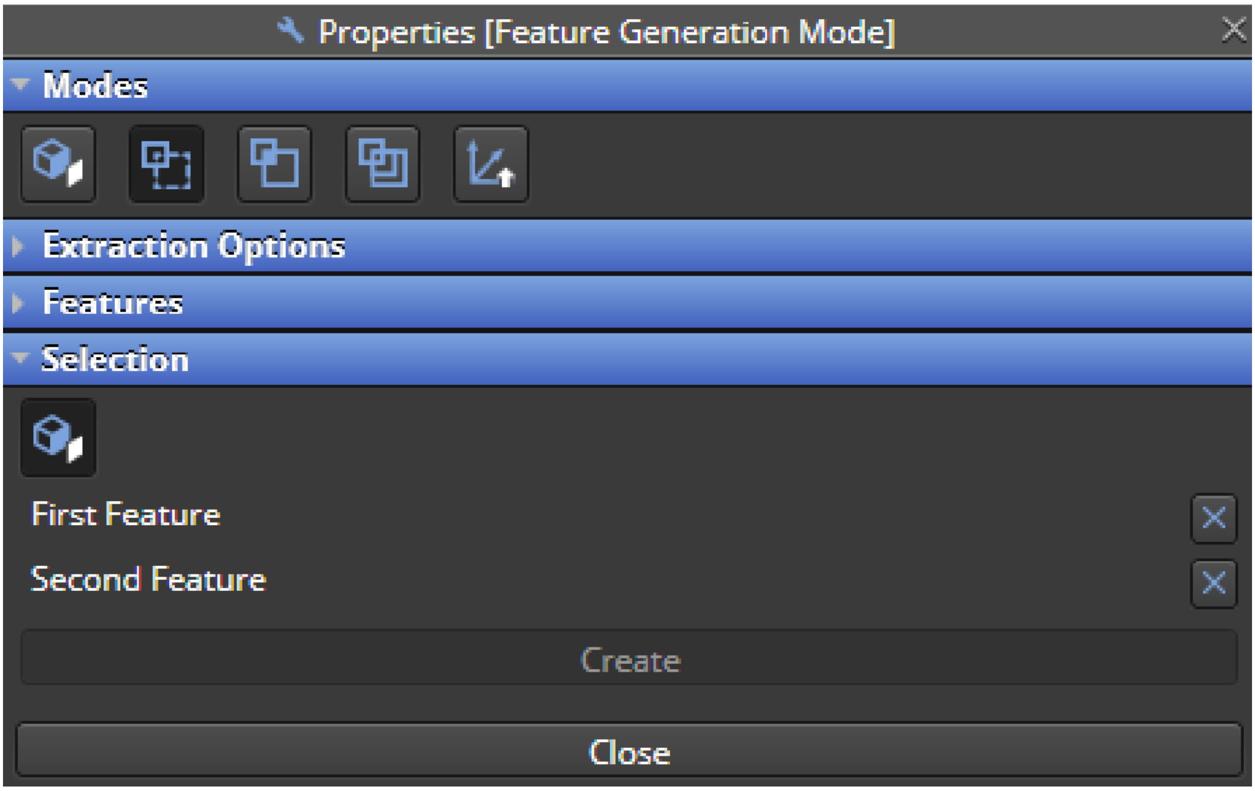
- Position: X 520.84903, Y 1913.32805, Z 1513.40211
- Rotation: Rx -89.80912, Ry -0.01864, Rz 2.12589
- Normal: i -0.03710, j 0.99931, k 3.33141E-003

The feature can be extracted from the 3D view with different options:

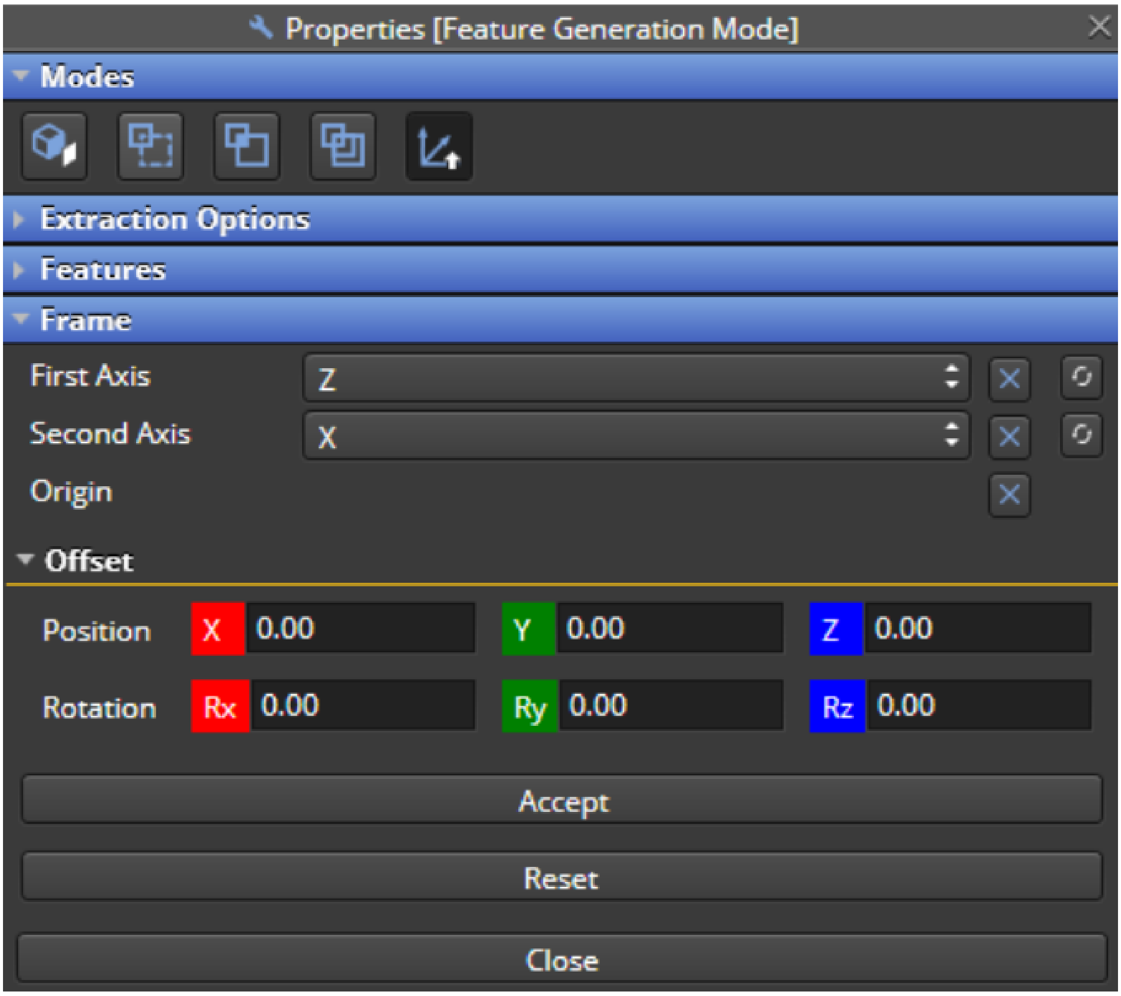
a) Feature
b) Feature From Points
c) Surface
d) Edge
e) Surface Point
f) Edge Point
g) Flanged Feature Mode



2) **Construct**: the user needs to select the features to do the construction as shown in the image below.
3) **Intersect**: the user needs to select the features to do the intersection as shown in the image below.
4) **Project**: the user needs to select the features to do the projection as shown in the image below.



5) **Frame**: the user needs to select axis and origin and the frame offset.



Properties [Feature Generation Mode]

First Axis: Z

Second Axis: X

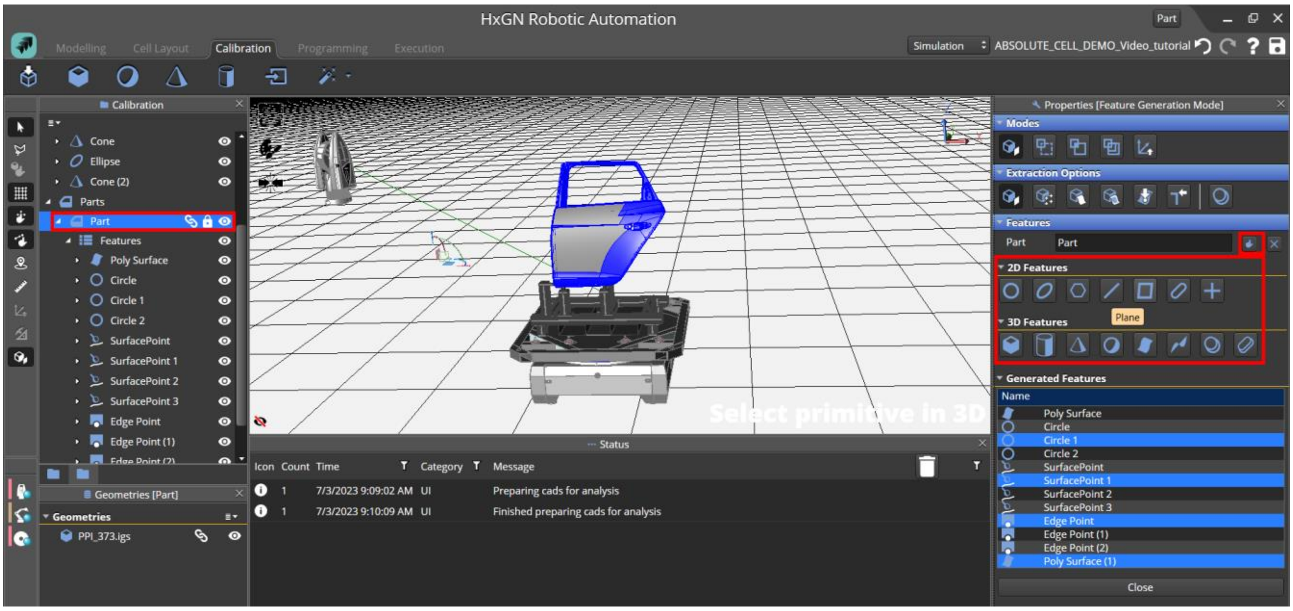
Origin

Offset

Position: X 0.00, Y 0.00, Z 0.00

Rotation: Rx 0.00, Ry 0.00, Rz 0.00

It is also possible to select a part in the project tree and assign explicitly features to it by clicking on the relative icon within the Features area.



Icon	Count	Time	Category	Message
●	1	7/3/2023 9:09:02 AM	UI	Preparing cads for analysis
●	1	7/3/2023 9:10:09 AM	UI	Finished preparing cads for analysis

7 Calibration and Alignment

This section is aimed for **Application Engineers**. Users do not have to calibrate robotic cells; they usually start using HxGN Robotic Automation already configured to work on the robotic cell installation where it is installed with the cell calibrated.

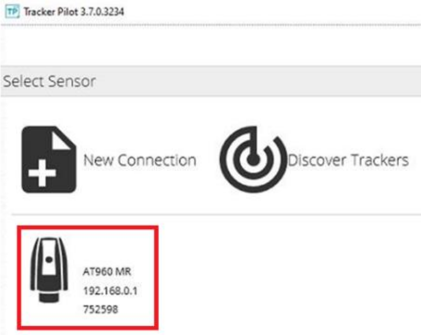
Cell Alignment is a critical step of performing the fine tuning of the cell model that was created virtually, and its goal is to match real cell with virtual cell.

Perquisites:
1) Leica Metrology Foundation (LMF) and Tracker installed on the system (AT 960-MR)
2) A Reflector (target) Type RRR (Red Ring Reflector) 1.5" (Check if it has tooling and its type).
3) Cell model imported to the online Robotic Automation system.

The cell alignment will use the tracker as a reference for aligning the cell, therefore, make sure to position the Tracker in X, Y, Z coordinates as much as possible to match the real cell.

Alternatively, you can position it in 0,0,0 coordinates and make sure not to move it during the complete process to complete alignment of all cell items. In this case the origin of the tracker reference system will coincide with the World reference system.

Execute the Tracker Pilot software in the PC, select the Sensor and obtain the IP address. If not visible, clicking on Discover Trackers will show the available tracker information.



Tracker Pilot 3.7.0.3234

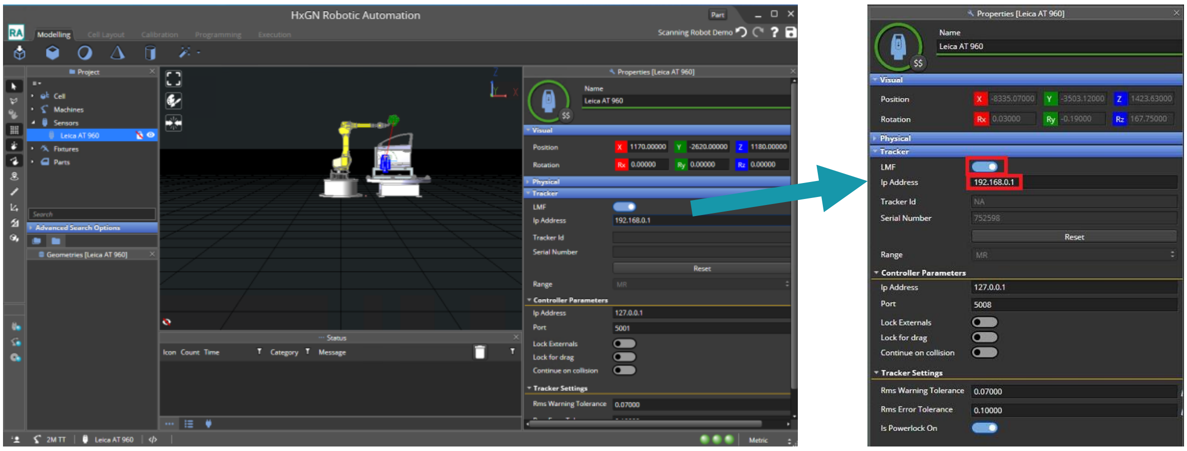
Select Sensor

New Connection

Discover Trackers

AT960 MR
192.168.0.1
752598

Open HxGN Robotic Automation application and import the cell model that needs be aligned.
Double click on the Leica AT960, update the IP address, and enable the LMF with the relative toggle switch:



Properties [Leica AT 960]

Name: Leica AT 960

Visual

Position: X -8335.07000 Y -3503.12000 Z 1423.63000

Rotation: Rx 0.03000 Ry -0.19000 Rz 167.75000

Physical

Tracker

LMF: on

Ip Address: 192.168.0.1

Tracker Id: NA

Serial Number: 752598

Reset

Range: MR

Controller Parameters

Ip Address: 127.0.0.1

Port: 5008

Lock Externals: off

Lock for drag: off

Continue on collision: off

Tracker Settings

Rms Warning Tolerance: 0.07000

Rms Error Tolerance: 0.10000

Is Powerlock On: on

7.1 Manual alignment workflow

The manual alignment process is done clicking on *Open Alignment Mode* under the Special Mode Toolbar (see chapter 4.3.7).

This process can be used to align any cell items, machines, sensors, fixtures and parts in the project and allows the user to manually select and align them in the 3D View.

During the alignment process the user needs to:

1) measure the real position of any item in the project
2) align the virtual item positions in the project to the real ones

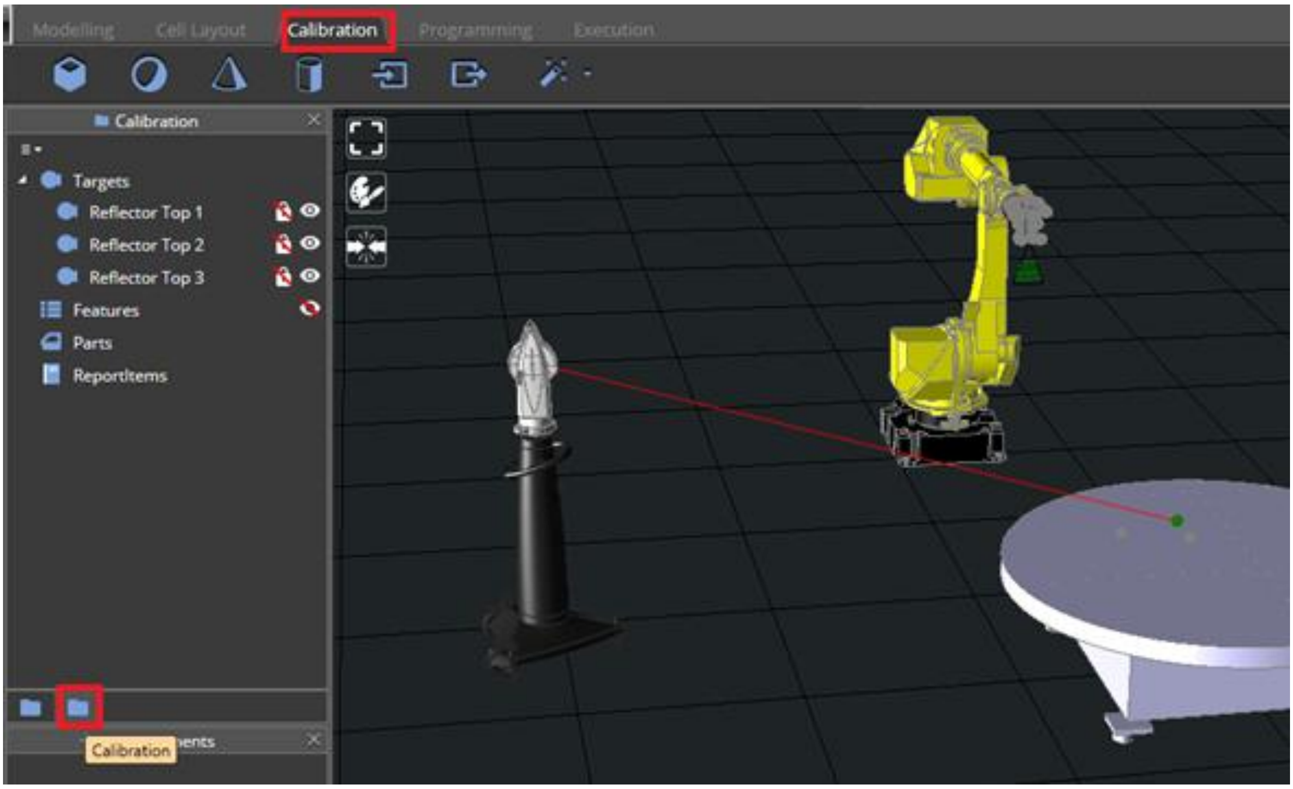
To be able to measure the real item positions, the user needs to connect to the cell tracker and get the points to construct the required feature with the help of any reflector or T-Probe device.

Depending on the cell configuration, the user will have to deal with one or more trackers, which are explained in the next chapters. This chapter covers the alignment workflow for single tracker cells. If the cell configuration contains more than one tracker, please refer to chapter 7.3.

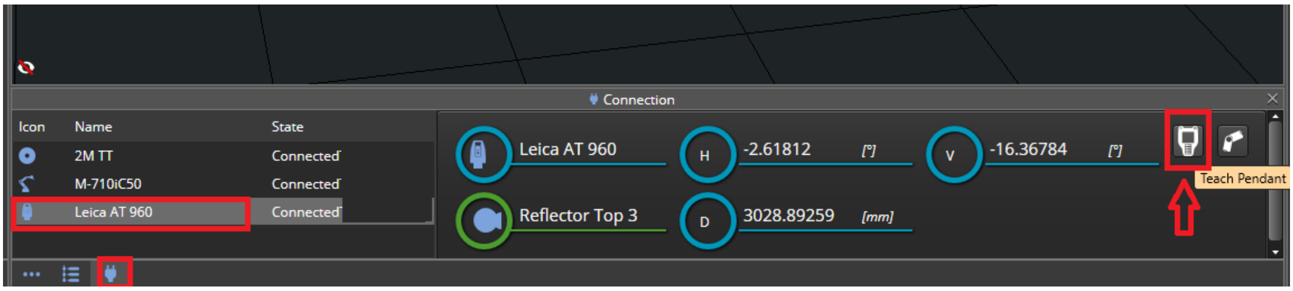
Working with one tracker simplifies the solution as all the measured features are in the same tracker coordinate system.

7.1.1 Turn Table alignment using three planes

In this use case, reflectors are used to align the turn table so they are placed on it.



Connect tracker with LMF and open the tracker Teach Pendant, then lock the Tracker on the first Reflector position. In this case, first plane will be the Turn Table plate and it is recommended to measure at least three points trying to cover the maximum surface as possible to minimize the error.



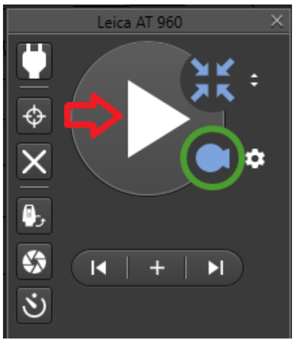
Connection

Icon	Name	State
	2M TT	Connected
	M-710iC50	Connected
	Leica AT 960	Connected

Leica AT 960 H -2.61812 [°] V -16.36784 [°]

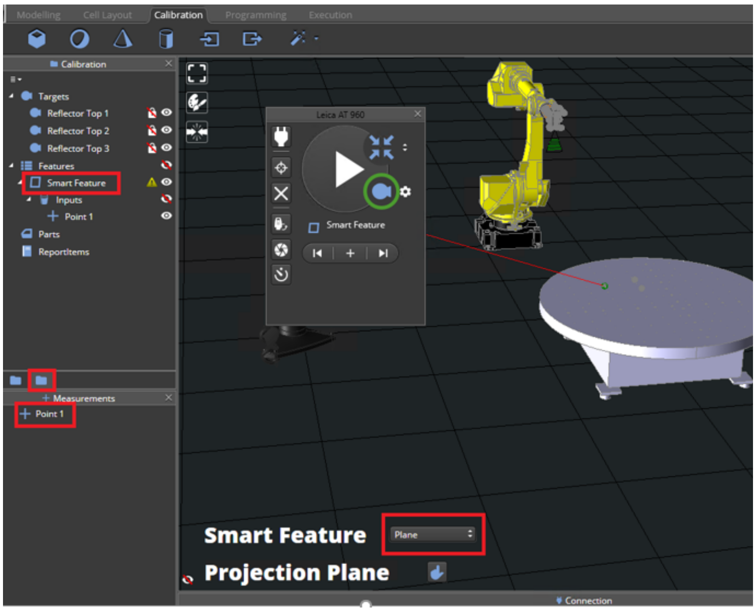
Reflector Top 3 D 3028.89259 [mm]

Click on the Measure button from the tracker teach pendant:



A Smart Feature will be created (**Point 1**), make sure to change its type to **Plane** from its drop-down menu.

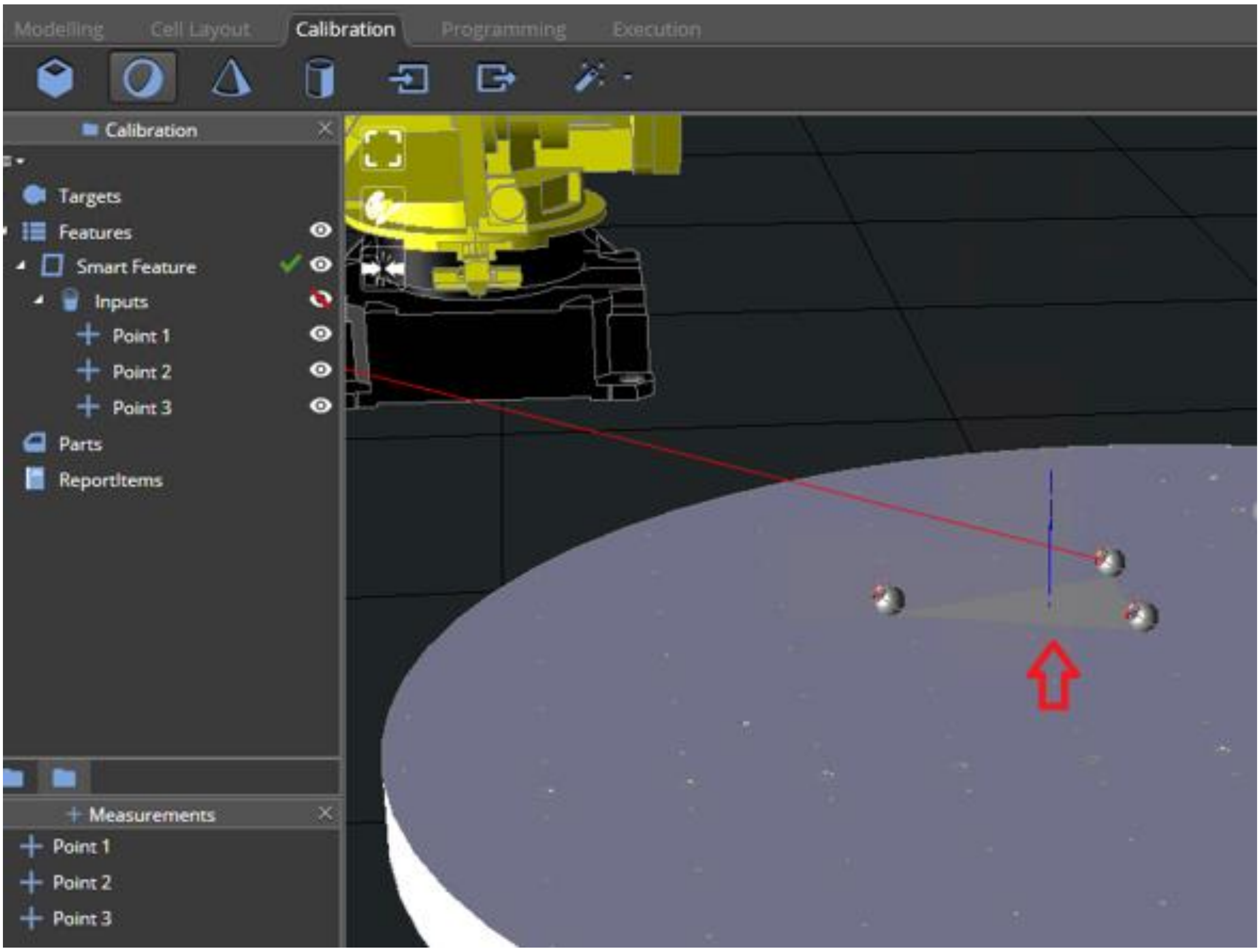
Info: Smart features are special features that allow the user to collect measurement points (inputs) within a group that automatically identify its type (e.g. two points are needed for a smart feature of type line, three points are needed for a smart feature of type plane, etc). The type can be also specified by the user using the relative drop-down menu in the properties.



Smart Feature Plane

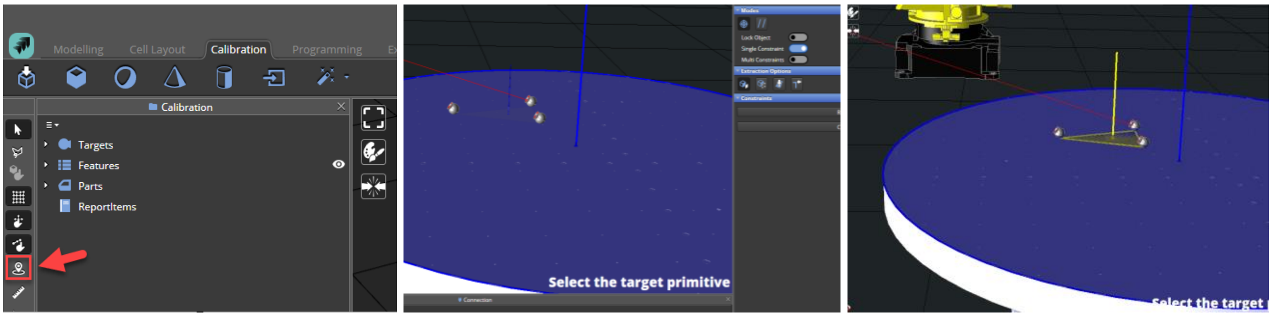
Projection Plane

Lock on the Tracker on second and third Reflector and repeat the measure to get three points, making sure it is computed ✓.

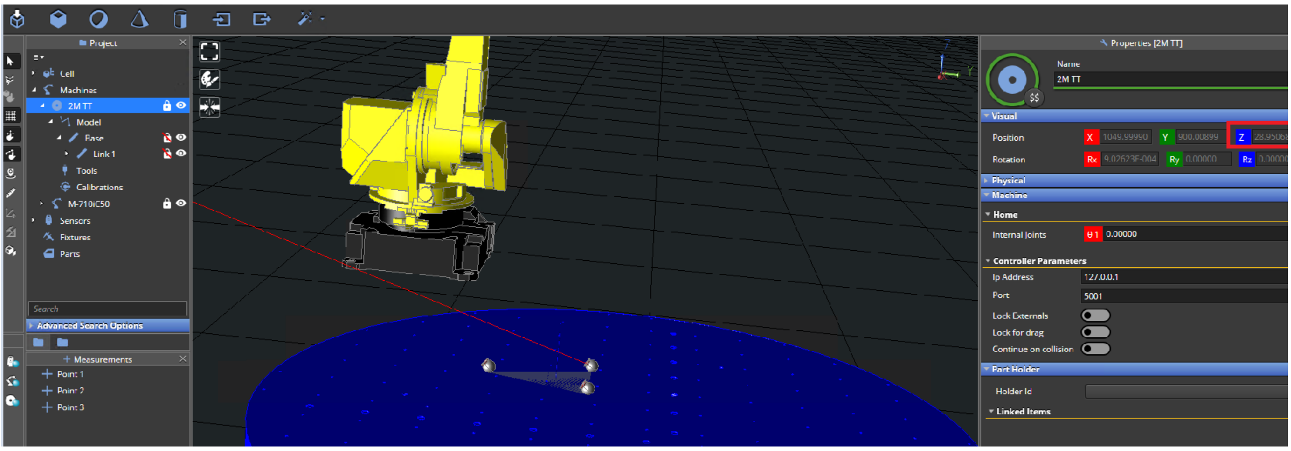


Double clicking on the Smart feature, it is possible to rename it to Top Plane, and verify if the plane setting is set to Below, as described in chapter 7.2.1.

Open the Alignment Mode and align the Top of the Turn Table CAD to the Top Plane feature you created before by first selecting the Turn Table CAD that you want to move first, and then selecting the target.



The Turn table cad will align with the top plane, and therefore its reference frame will be moved to match the real one that was measured by the three reflectors (in this example only the Z axis will be changed in a relevant way).



Name: 2M TT

Position: X 1049.99990 Y 900.00899 Z 28.95068

Rotation: Rx 9.02623E-004 Ry 0.00000 Rz 0.00000

Internal Joints: θ1 0.00000

Ip Address: 127.0.0.1

Port: 5001

Lock Externals: ○

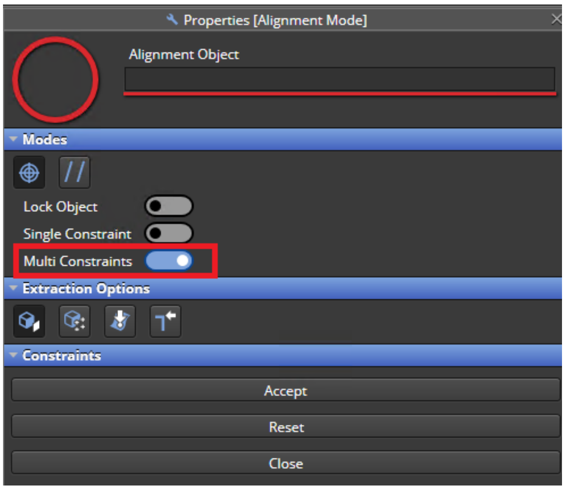
Lock for drag: ○

Continue on collision: ○

Holder Id:

Accept when done and repeat the process again with two more planes.

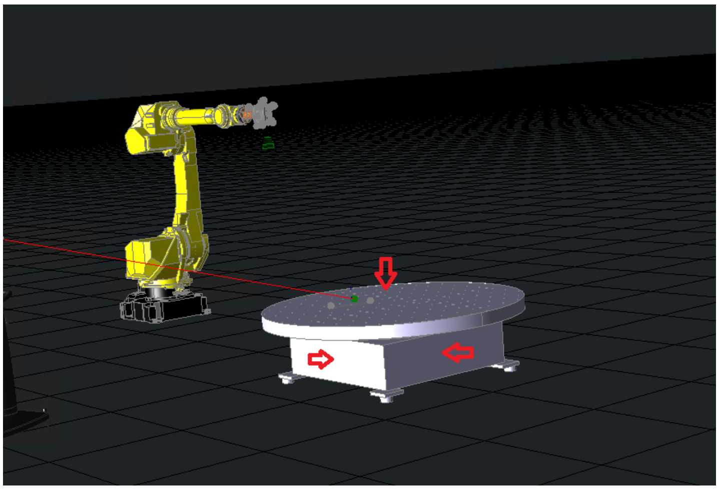
It is recommended to create at least three measured planes, and then perform an alignment using **Multi Constraints** option (other extraction options are explained in chapter 4.3.7.) by repeating the steps described above to align a plane three times, one for each plane to be measured.



Lock Object: ○

Single Constraint: ○

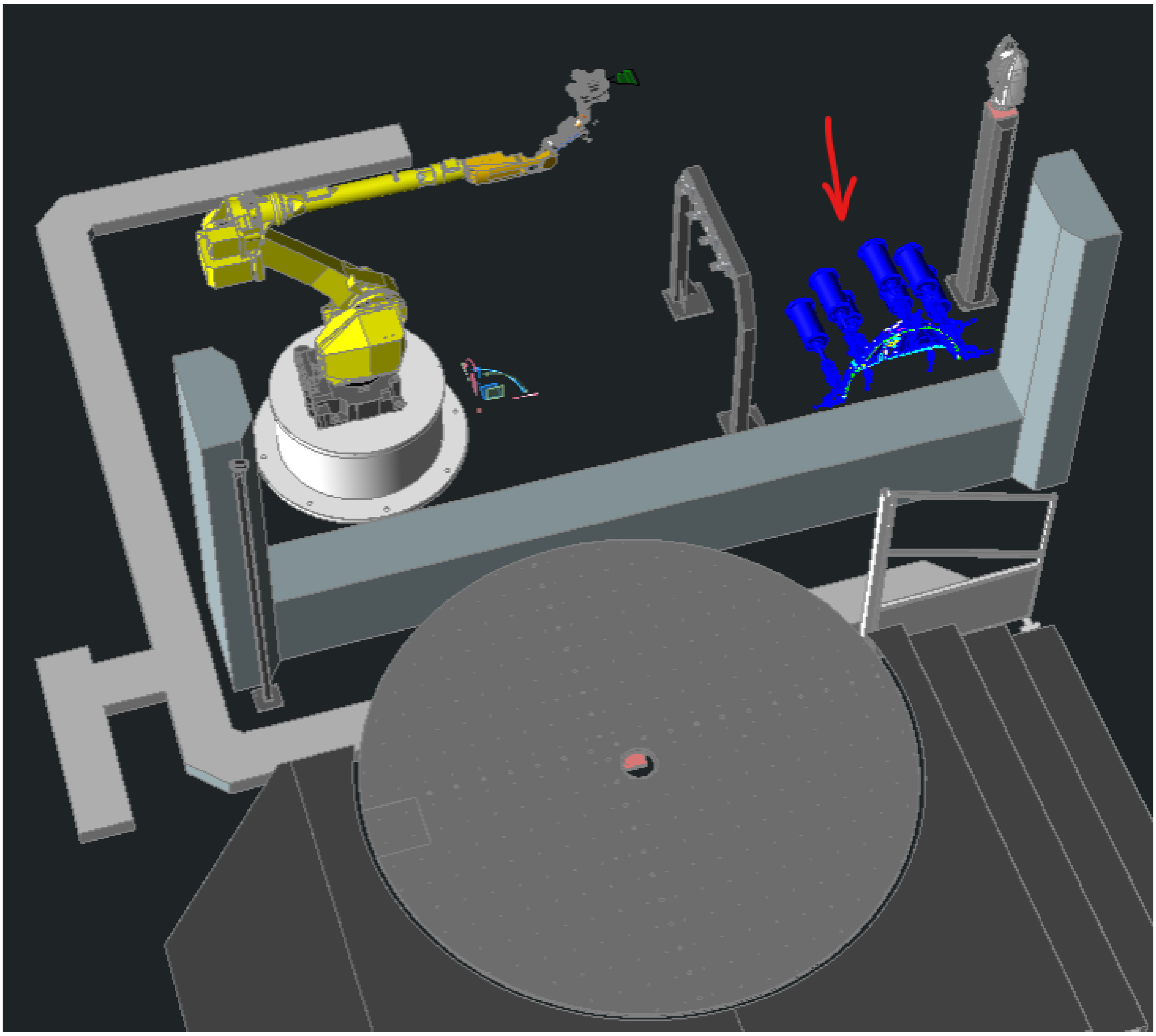
Multi Constraints: ●



7.1.2 Part alignment using features

In this use case, the manual alignment is used to align the part to the fixture. The fixture and the part can be added to the cell as described in chapter 5.2.1.5 and 5.2.1.6 respectively.

After adding the fixture and the part, they can be moved to the correct position in the cell. In most cases the imported part will be positioned in the right location in relation to its fixture but they both will be in their original design coordinates (e.g. car coordinate system).

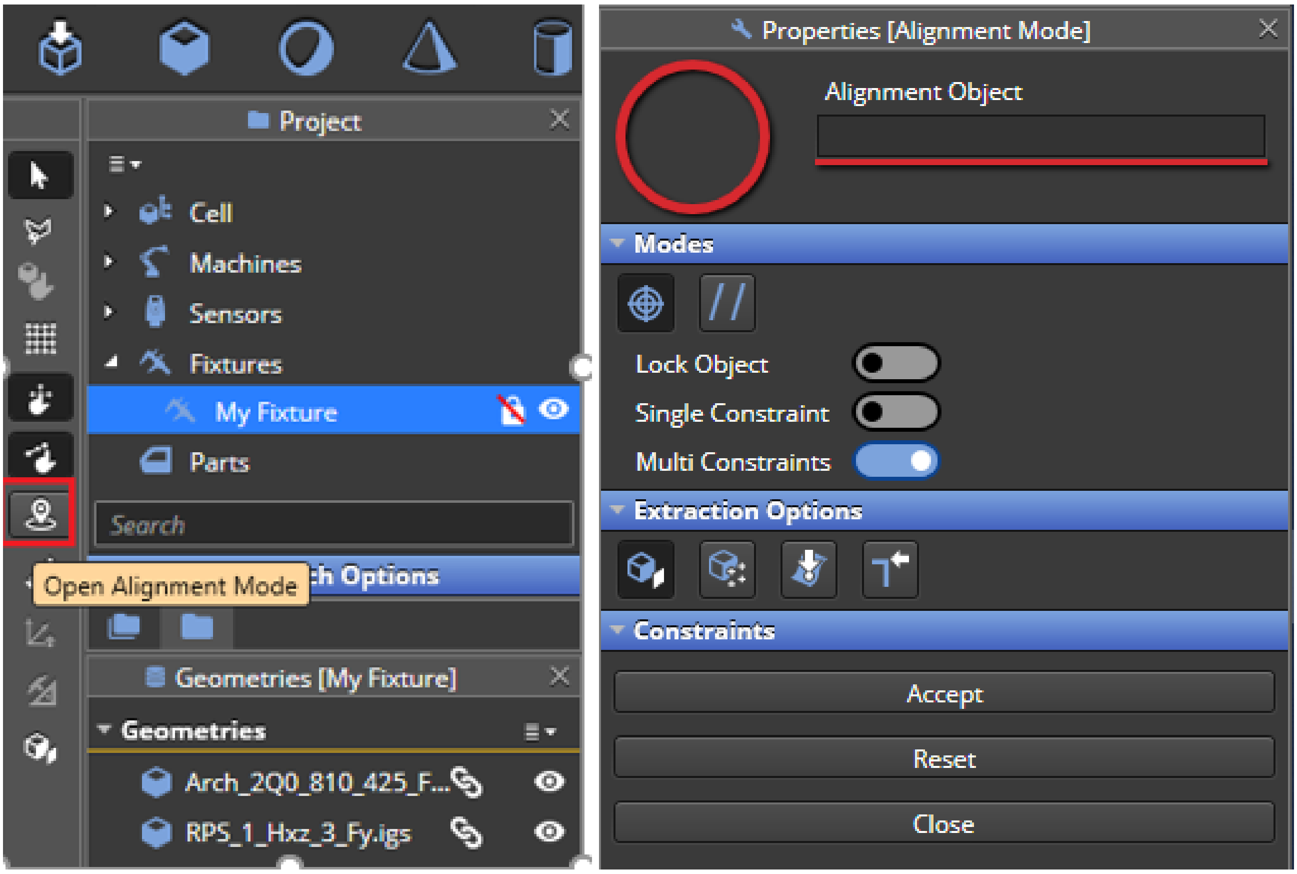


The process of aligning the fixture in the cell consist of selecting geometries on the fixture and align them with the corresponding geometries in the real cell. The fixture coordinate system will be aligned to the reference frame of the corresponding geometries.
There are two types of corresponding geometries:

- **Nominal CAD geometries** – This option is used when the fixture has a distinct position in the cell. In our example, the distinct position is shown by holes in the turntable into which the fixture columns will be fitted. Also in this case is suggested to measure the actual position of the nominal geometries.

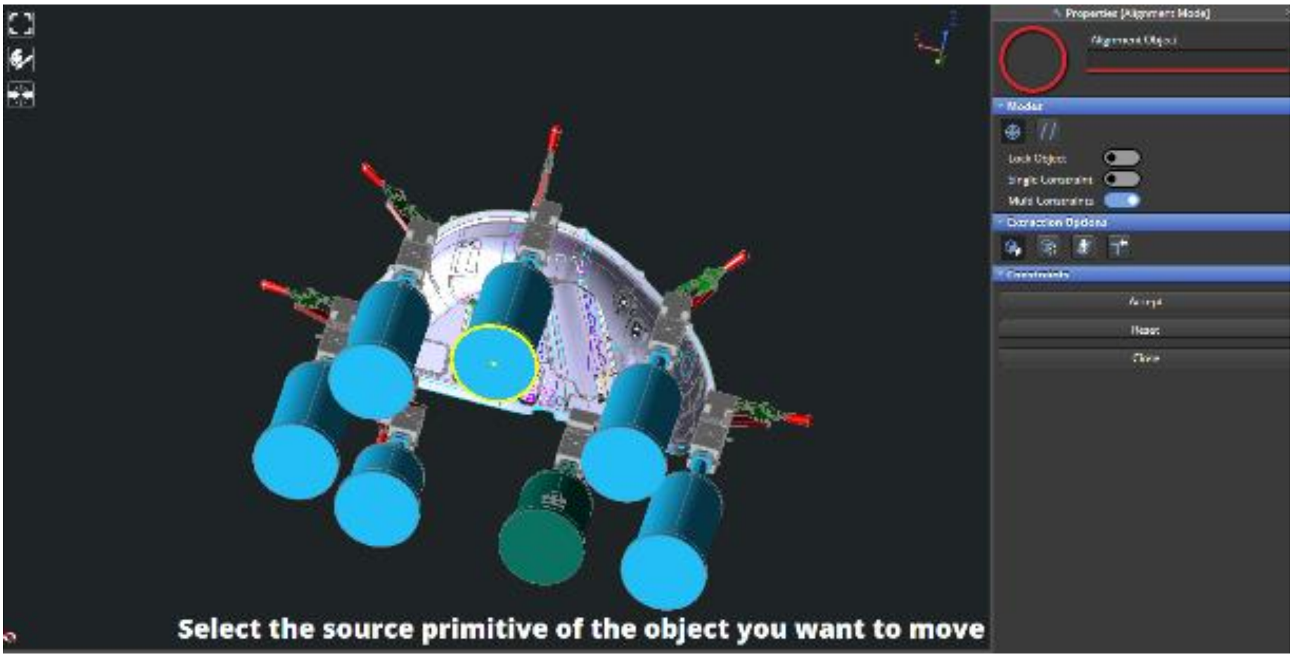
- **Actual geometries** – These are geometries that must be measured on the real fixture in the cell. In most cases you would use a T-Probe or a reflector to measure these geometries. To learn more on how to measure actual geometries refer to the section 4.3.8 on how to measure cell geometries.

After opening the Alignment mode, in the Properties it is needed to switch on the **Multi Constraints** mode. Other extraction options are explained in chapter 4.3.7.

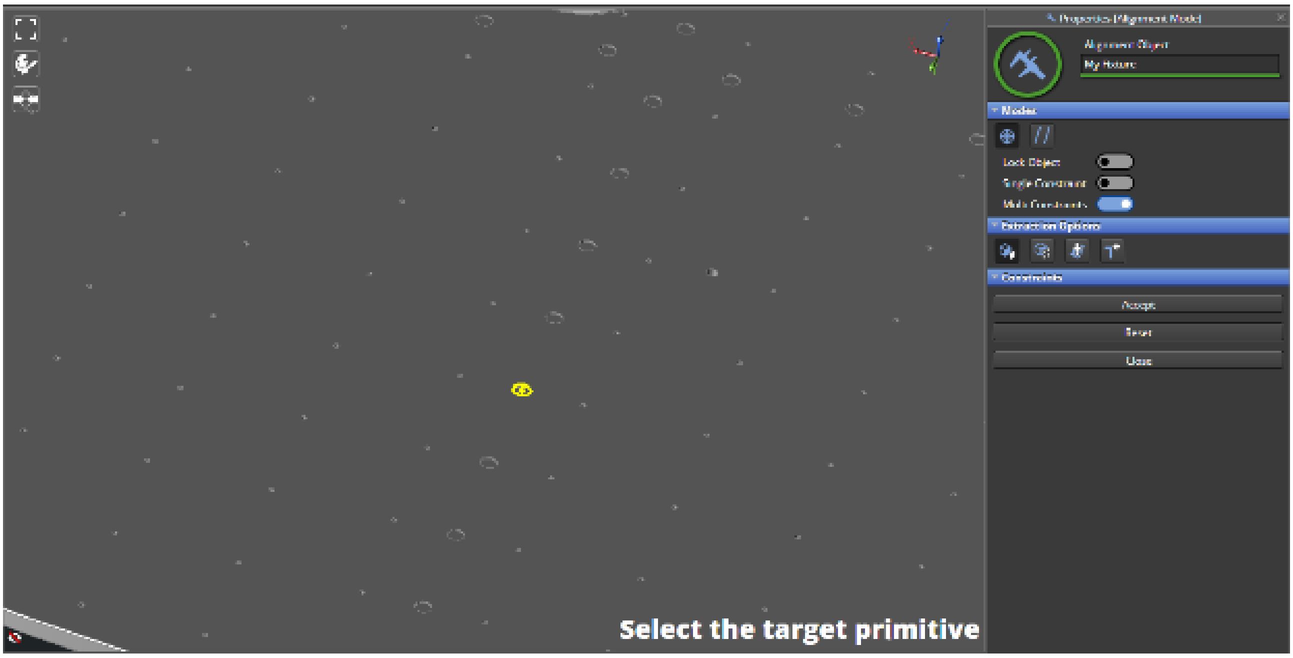


Lock Object ○
Single Constraint ○
Multi Constraints ●

It is possible to proceed by picking a feature on the element which will move (fixture) and then pick the corresponding element on the destination (Turn table), continuing picking elements on the fixture and on the turntable to set all the alignment constraints.

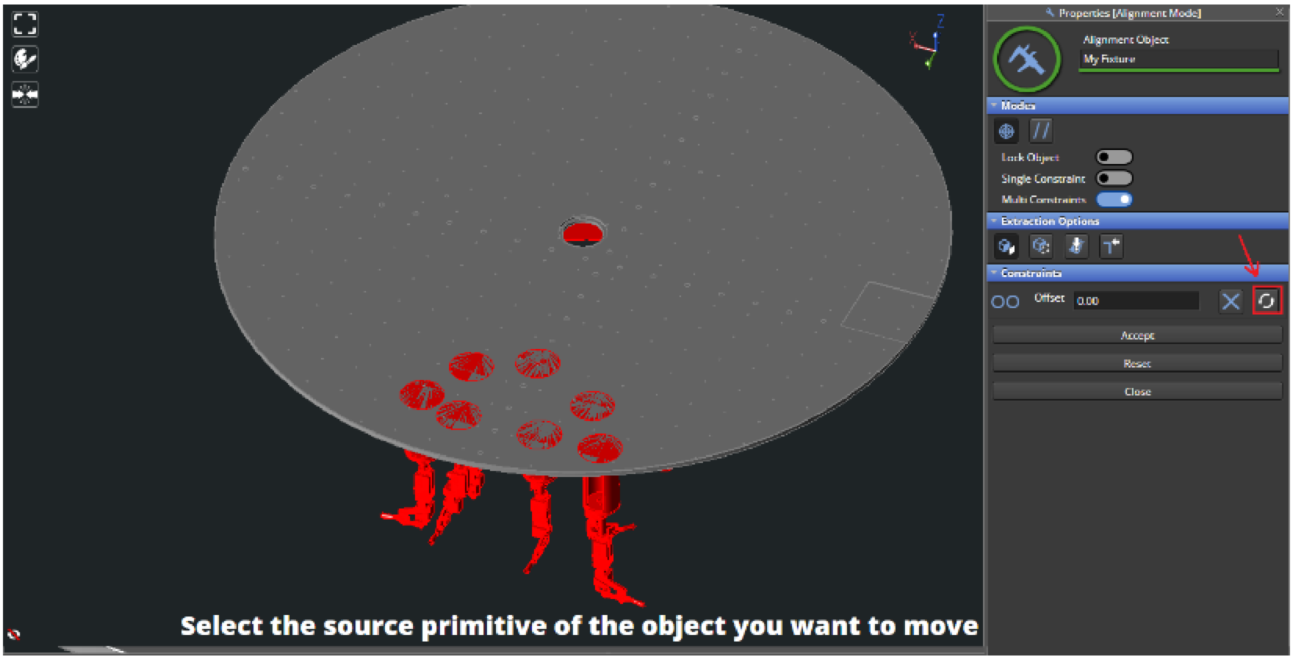


Select the source primitive of the object you want to move



Select the target primitive

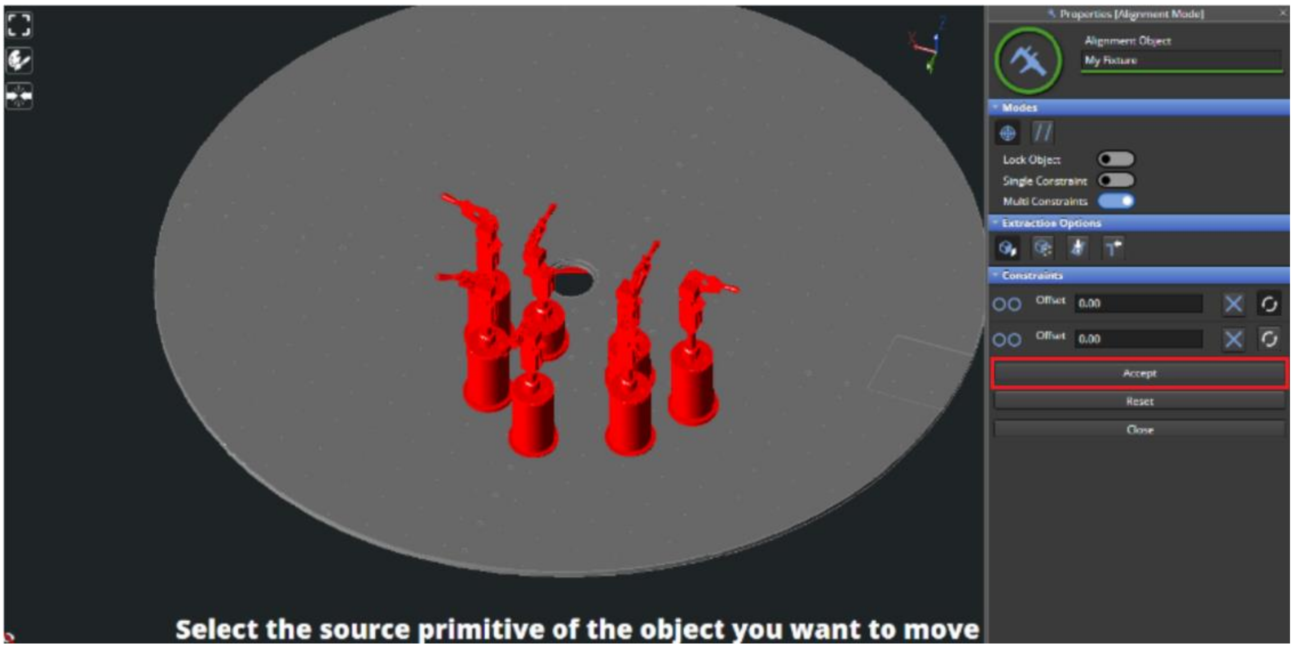
After clicking on the target primitive, the fixture will be aligned to the turn table position. In case the fixture is aligned in a flipped position, you can flip it back using the **Reverse** button.



Select the source primitive of the object you want to move

The user can continue adding other constraints and once the fixture is in the correct position, press on the **Accept** button, and close the alignment mode.

Info: In the following picture only two constraints are used but it is strongly recommended to use three.



Properties [Alignment Mode]

Alignment Object: My Fixture

Modes

Lock Object: off
Single Constraint: off
Multi Constraints: on

Extraction Options

Constraints

Offset 0.00
Offset 0.00

Select the source primitive of the object you want to move

In case you want to use actual geometries for the fixture alignment, you need to create them extracting from the CAD before you start the alignment process. In the alignment process, instead of selecting elements on the turntable as the destination, select the corresponding actual geometries.

The fixture is now in the correct position, but the part is not.
To align the part to the fixture, you can follow the same steps described above, selecting:

- the part elements instead of the fixture elements
- the fixture elements instead of the turn table elements.

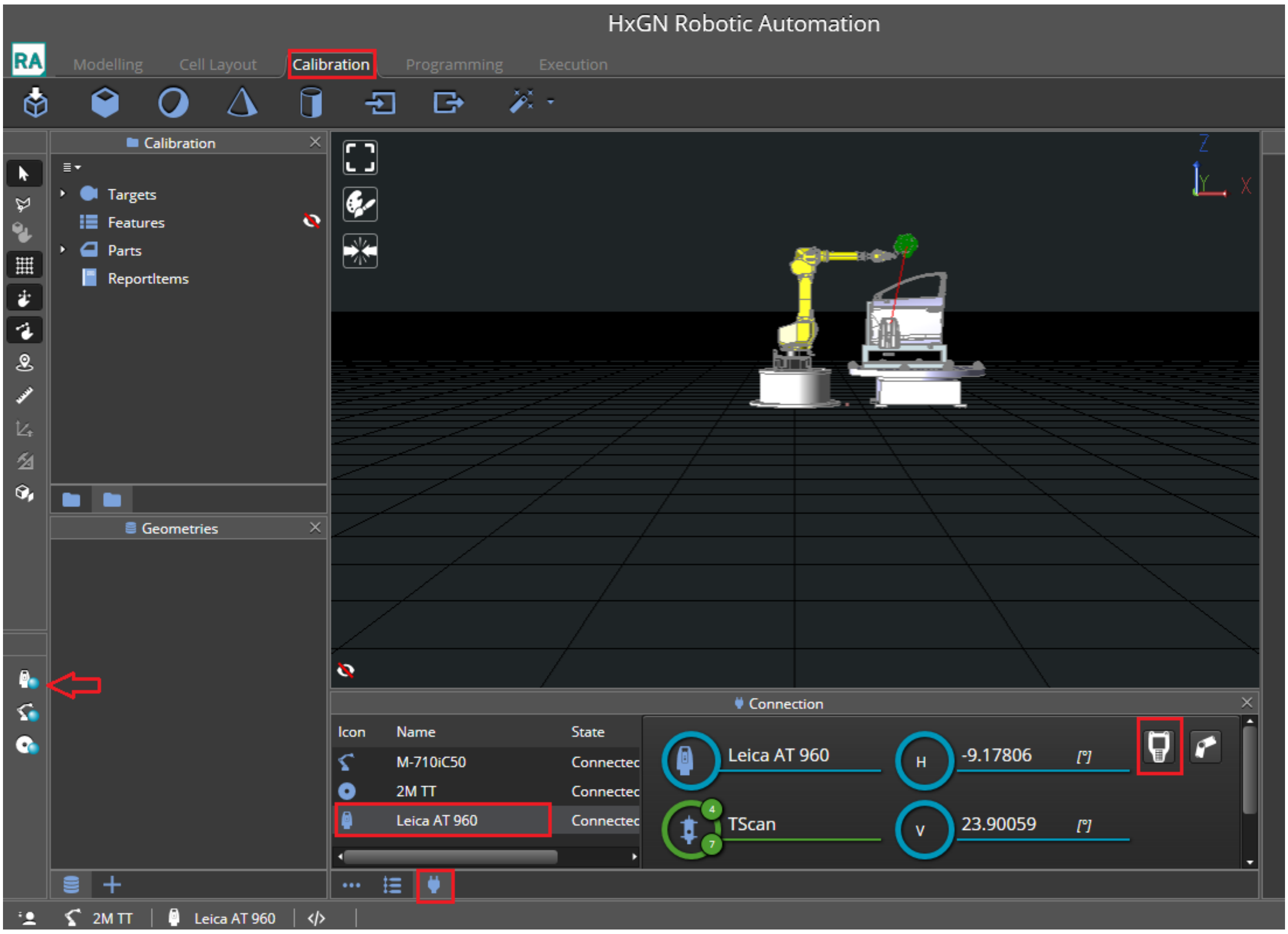
At the end of the alignment process the part will also be in its correct position so both the fixture and the part will be aligned to their real position in the cell.

At this point it is needed to lock the part and fixture positions as described in chapters 5.2.1.5 and 5.2.1.6.

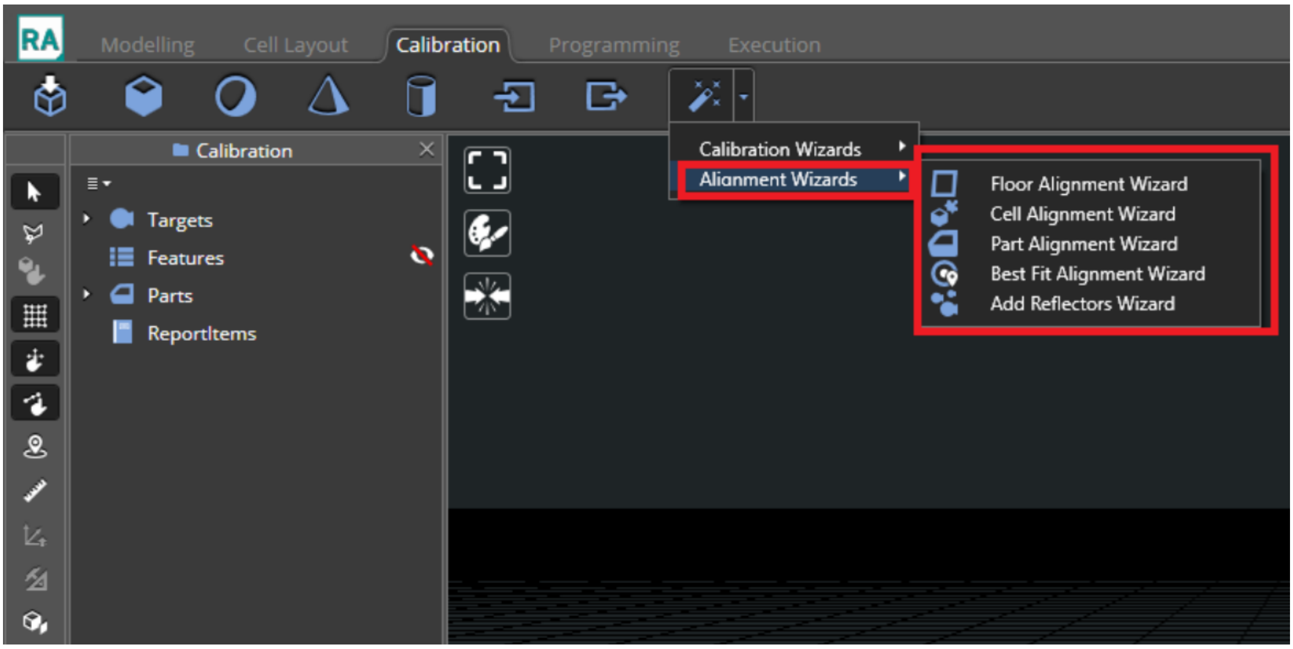
If you would like to place the part on the turn table without a fixture, you will still need a fixture object as you cannot assign the part directly to the turn table.
You can create a new fixture without assigning any CAD to it.
Next, drag the empty fixture onto the turntable, and the part onto the empty fixture.

7.2 Alignment wizards workflow

Alignment is done through wizards under the calibration tab.
Click on calibration mode tab and then click on connect. The LED will switch from blue to green:



Once you are in Calibration mode, Open the wizard panel on upper right. There is a wizard for each type of alignment.



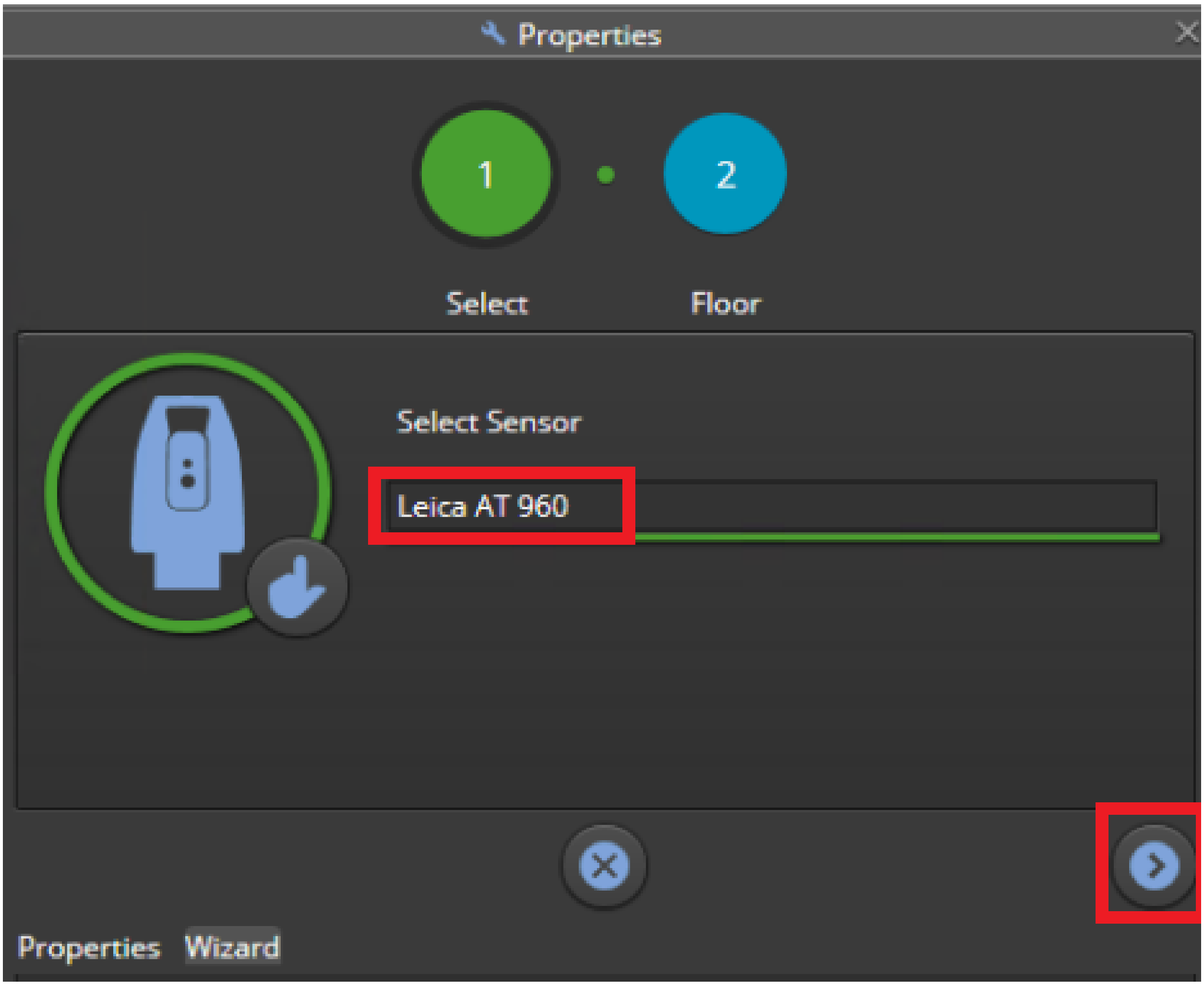
7.2.1 Floor alignment wizard

It is particularly important to start first with Tracker and Floor Alignment. The purpose of Floor Alignment wizard is to align the Tracker reference frame with the floor reference frame.

Info: it is suggested to perform the floor alignment within the complete Cell alignment wizard explained in the section 7.2.2.

The following steps are needed:

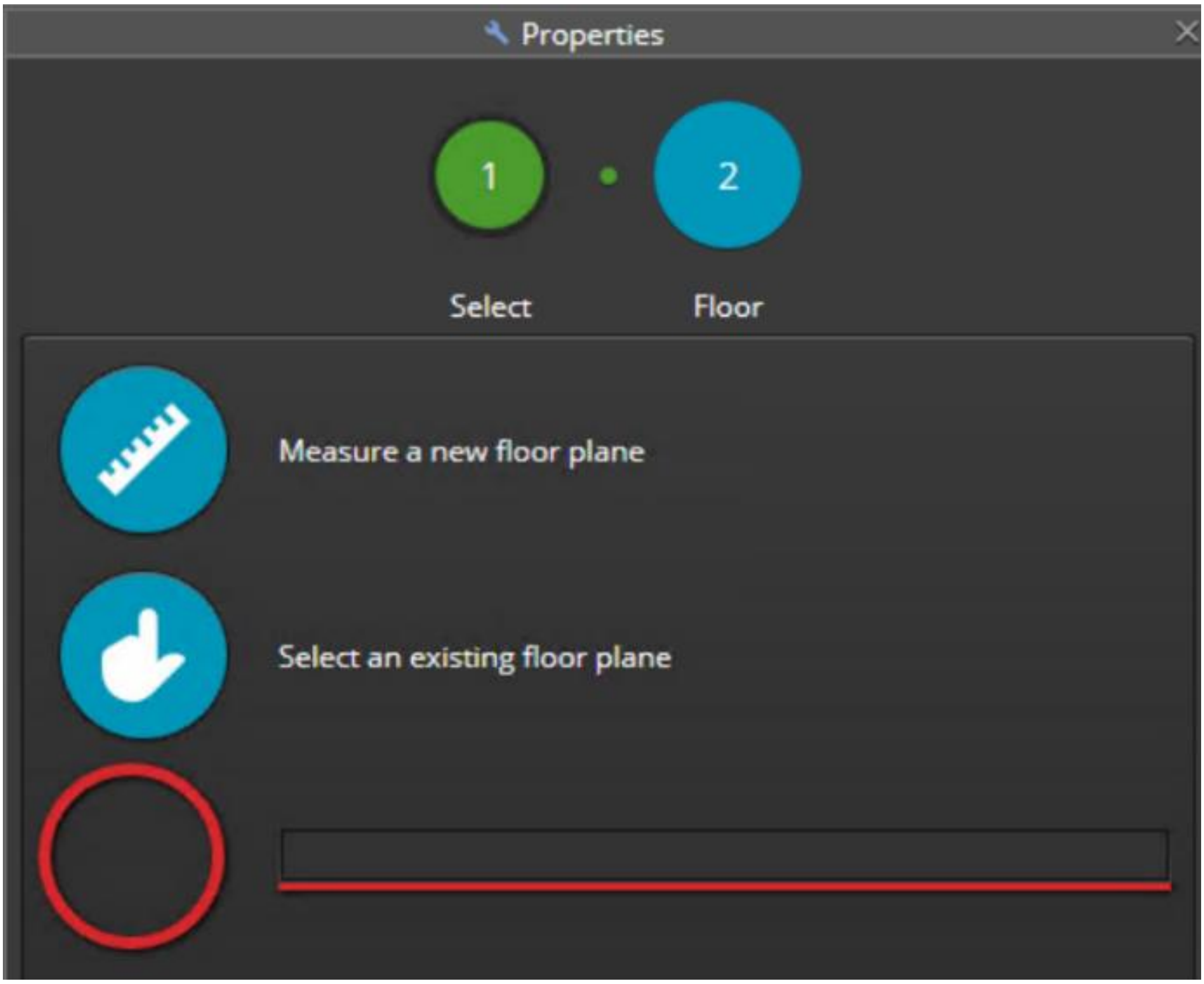
1) Select the Leica AT960 Sensor and click next to continue.



Select Sensor

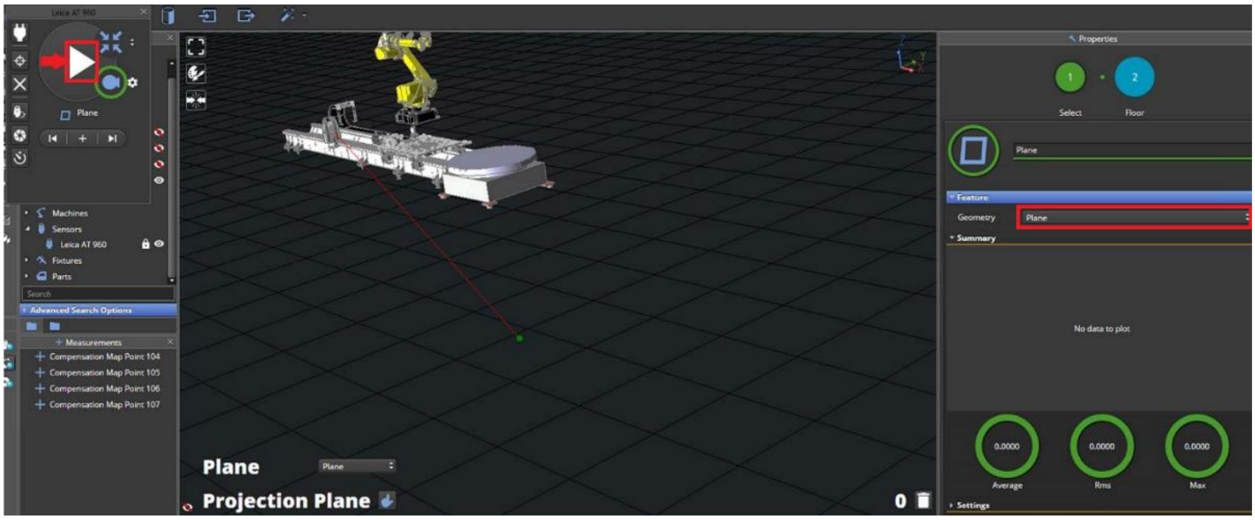
Leica AT 960

2) You may Measure a new Floor Plane or Select an existing Plane:
   a) **Measure a new floor plane** opens teach pendant to measure a plane feature.
   b) **Select an existing floor plane** selects a previously measured feature being the floor plane.

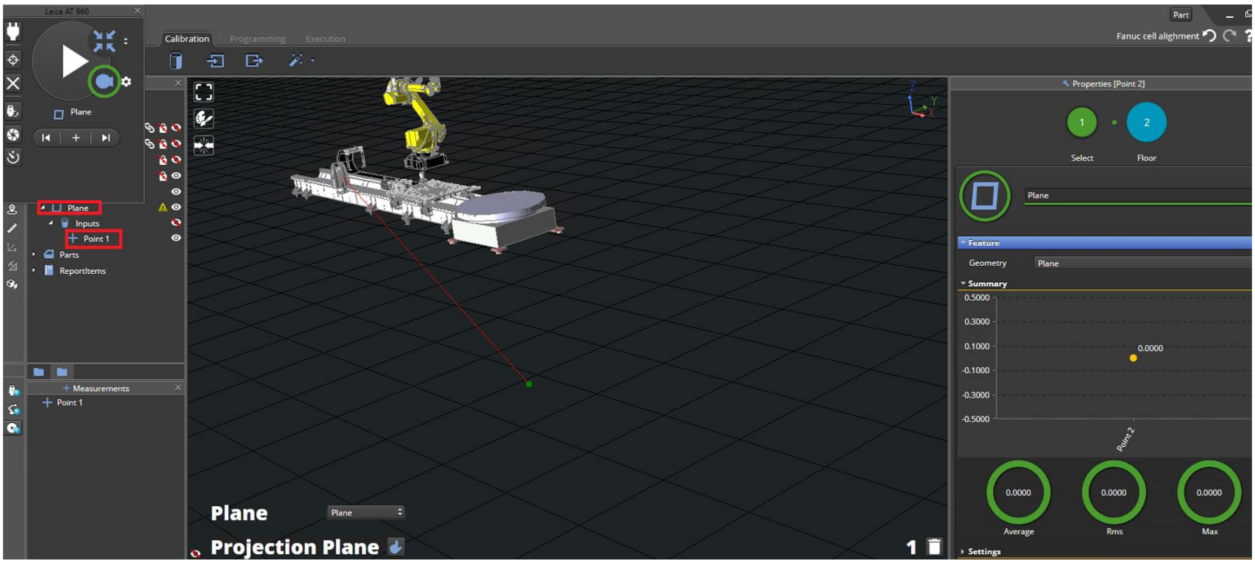


In the following we will continue selecting **Measure a new floor plane**

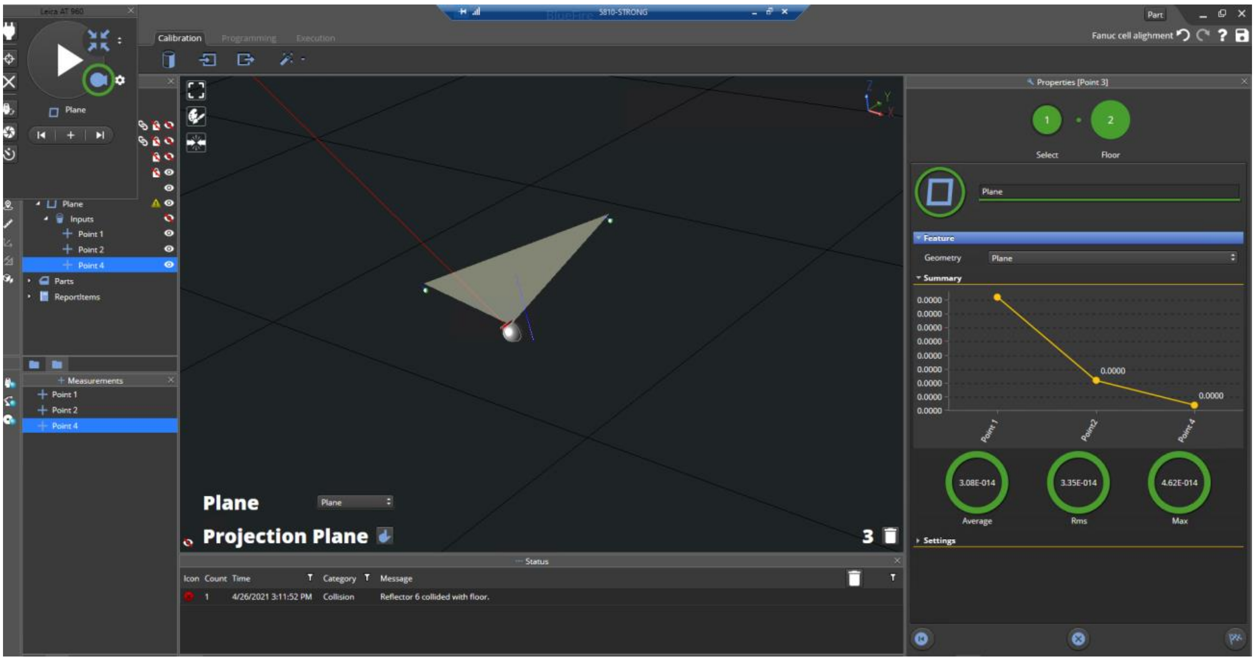
Locate the Reflector on the floor and lock the tracker on it.



The Teach pendant will open, select plane geometry on properties, click on the Play icon (▶) and measure at least three points to define a plane.
Once your click, you will see under *Features* | *Plane* | *Inputs* the point that was created (Point1).

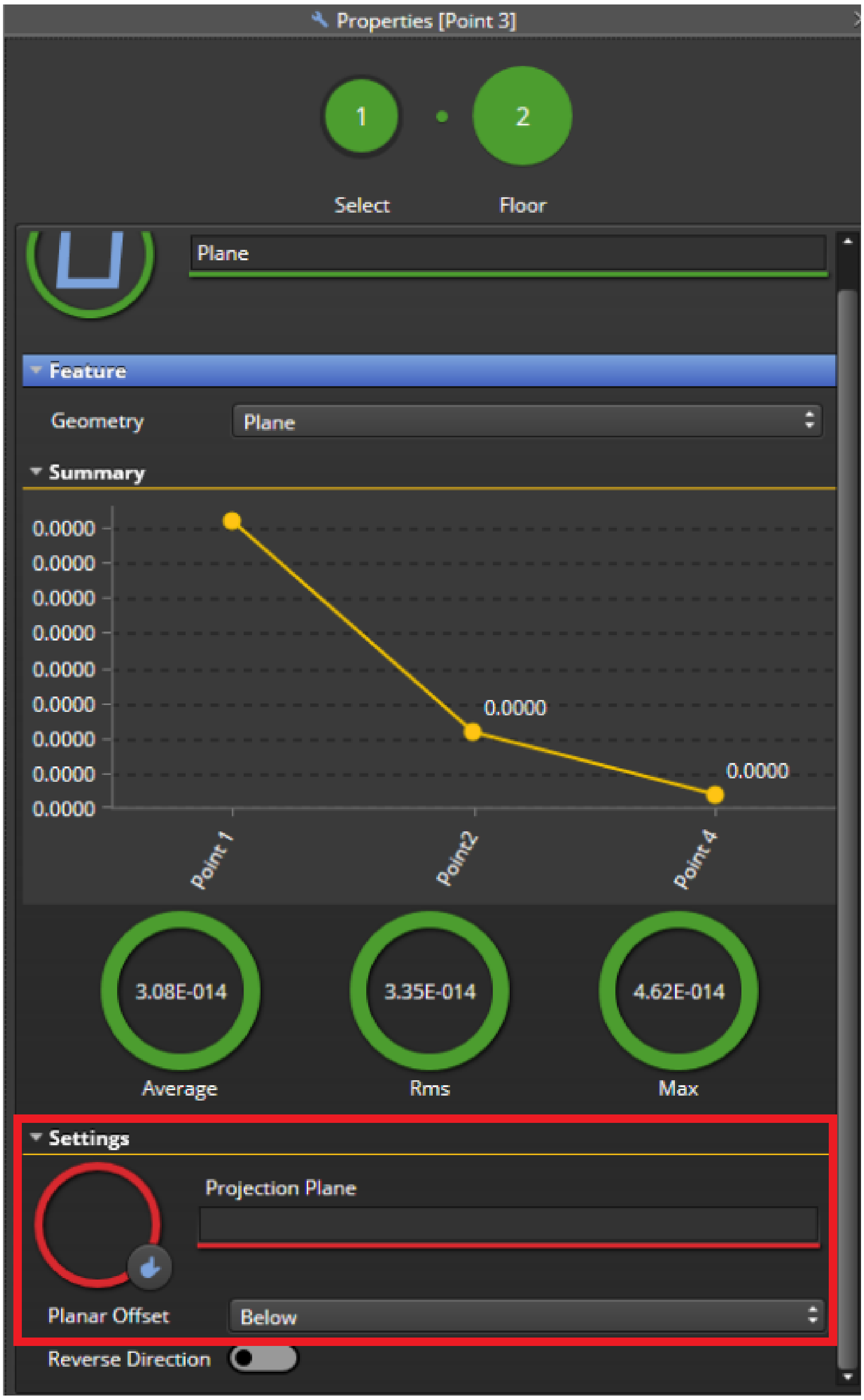


Move the Reflector to another position on the floor and click to measure again. Repeat again at least 3 points until you see the plane in the 3D Window.



Icon	Count	Time	Category	Message
	1	4/26/2021 3:11:52 PM	Collision	Reflector 6 collided with floor.

If you are using a tool to place the Reflectors on (e.g. a nest) and it is below the reflector, select **Below** from the drop-down menu in *Properties* | *Settings* | *Projection Plane* within properties.



Floor and plane have different heights. After clicking the Finish icon (▣) the plane height will be adjusted automatically to the floor grid, and the tracker Z height value will be adjusted to the actual height in the real cell.

It is recommended to record and write down that Z value for future reference.

7.2.2 Cell alignment wizard

The cell alignment wizard allows the user to align and calibrate items in the virtual cell to make it match the real cell, starting from the tracker to every cell item and create reference targets for best fit alignment.

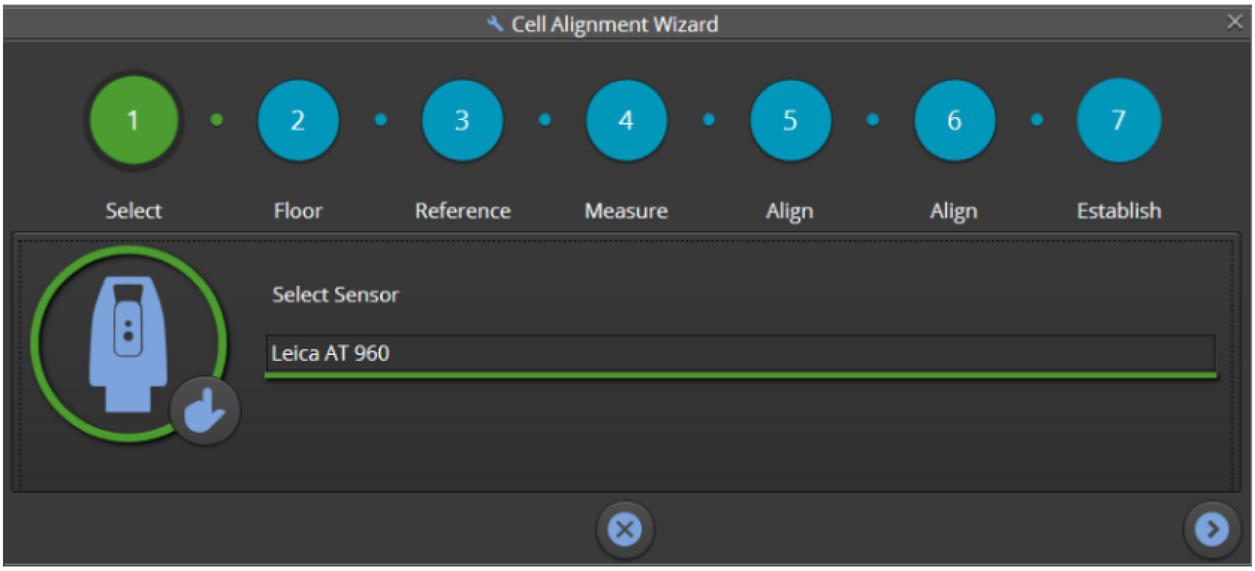
This wizard is designed for single tracker cell configurations.
For dual robot configurations, please follow the steps described in chapter 7.1.2.

This wizard currently forces the user to go through the following flow:

Floor** → **Reference** → **Measure** → **Align** → **Establish

Following the wizard steps:

1) **Select**: select sensor used to align the cell



Select Sensor

Leica AT 960

2) **Floor** step (if Floor alignment was done previously, it is possible to skip the *Floor* step described in section 7.2.1 and continue with the remaining steps):

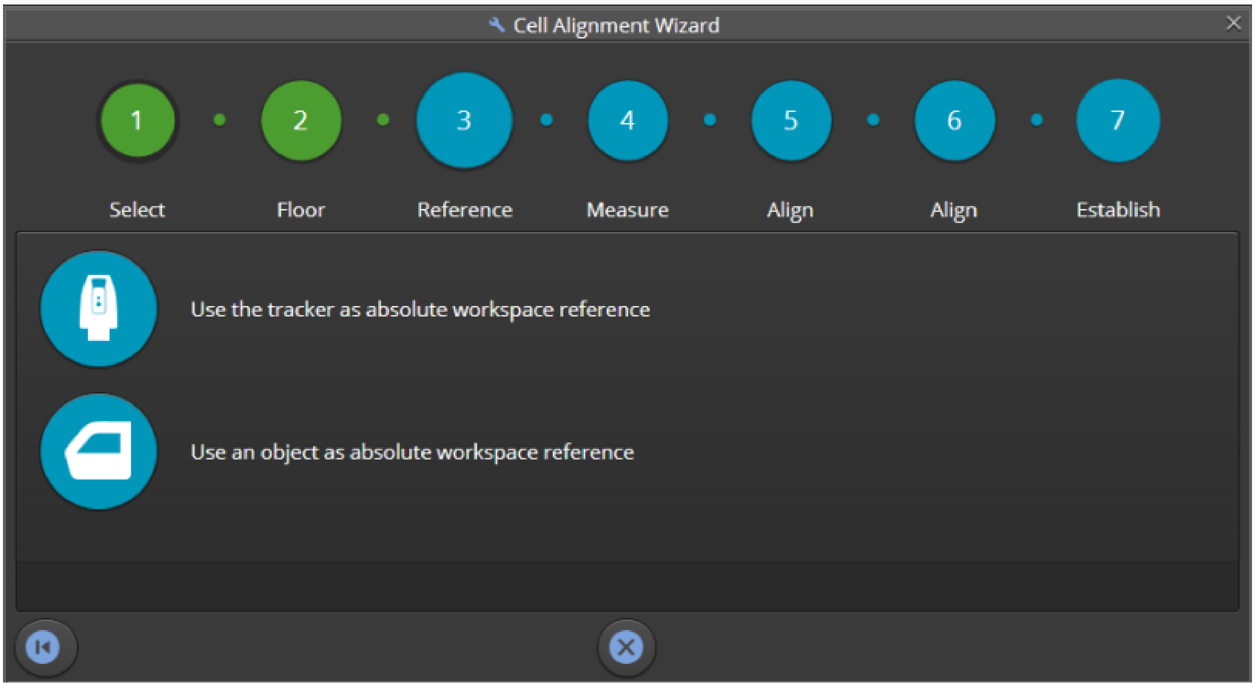
   a) **Measure a new floor plane** to open a tech pendant to measure a plane feature.
   b) **Select an existing floor plane** to choose a previously feature being the floor plane.

3) **Reference**: select the reference frame:
   a) **Use the tracker as absolute workspace reference**: cell alignment will use tracker as reference for aligning the cell.

   If tracker is going to be the reference object, it is a good practice to set position on (0,0,0).

   b) **Use an object as absolute workspace reference**: cell alignment will use an object as reference for aligning the cell.

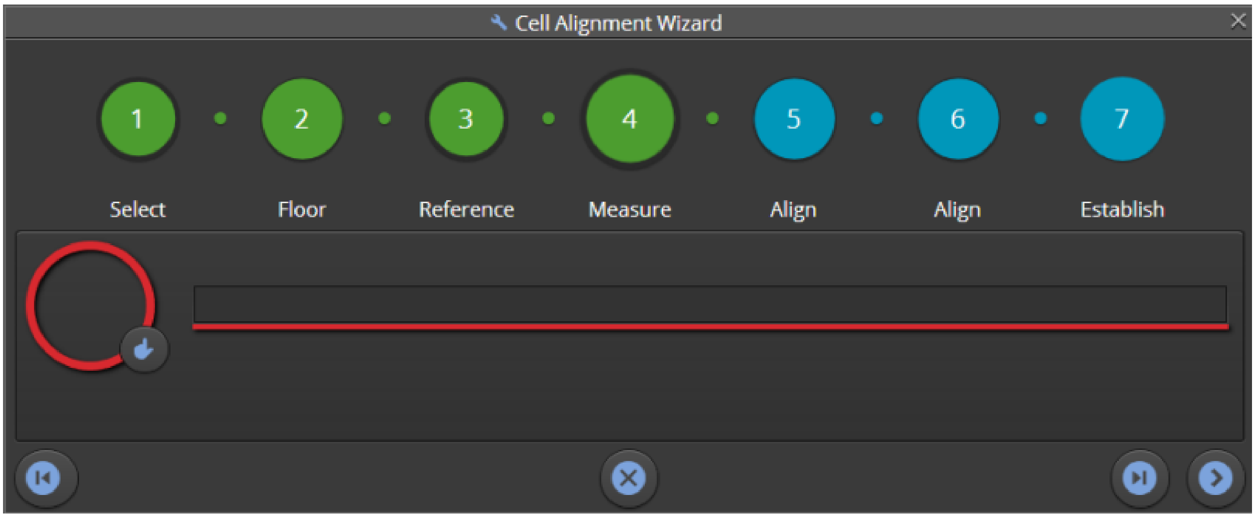
In case the Tracker is used as absolute workspace reference, the user can continue selecting the first option. If an object is used as absolute workspace reference , the user can select the second option. See chapter 7.2.2.1 to see how a Turn table is used as an absolute workspace reference.



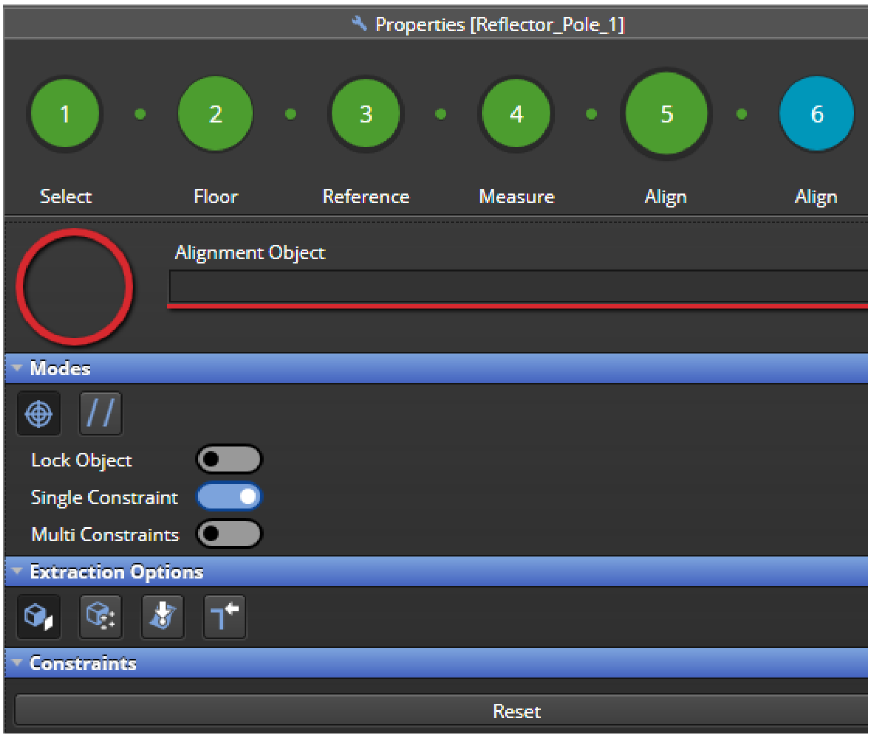
Use the tracker as absolute workspace reference

Use an object as absolute workspace reference

4) **Measure**: the tracker teach pendant will open to measure all the features considered by the user to align the cell items.



5) **Align**: this step is skipped in case the tracker is used and absolute workspace reference so there is no need to align the tracker to the object. If an object is used as absolute workspace reference, in this step the tracker will be aligned to the selected object, compensating for alignment error.



Properties [Reflector_Pole_1]

Alignment Object

Modes

Lock Object ○

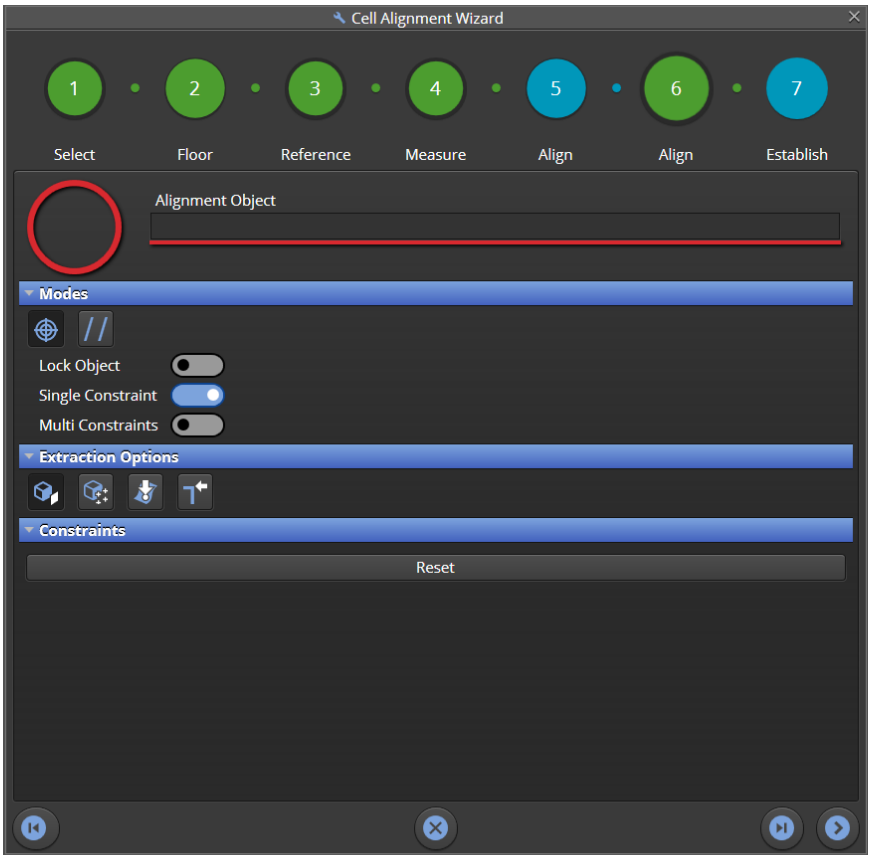
Single Constraint ●

Multi Constraints ○

Extraction Options

Constraints

6) **Align**: in this step, it possible to align all items in the cell to the absolute workspace reference that was defined in the Reference step (3).



Cell Alignment Wizard

Alignment Object

Modes

Lock Object ○

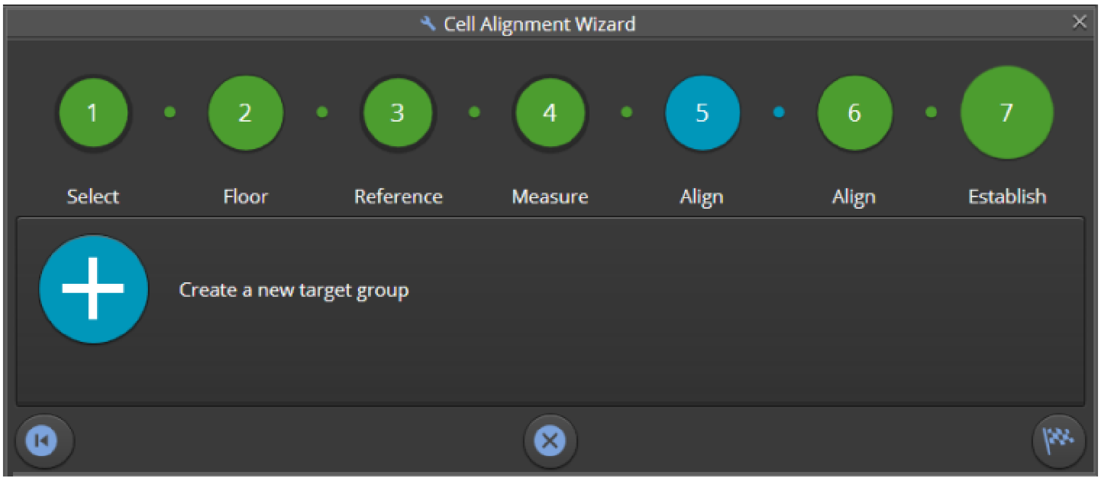
Single Constraint ●

Multi Constraints ○

Extraction Options

Constraints

7) **Establish:** in this step it is possible to define a target group of reflectors (at least three are required, see chapter 7.2.5) to be used in Best Fit alignment wizard to re-align the tracker in case of movement from the previous location (e.g. in case of maintenance or replacement).

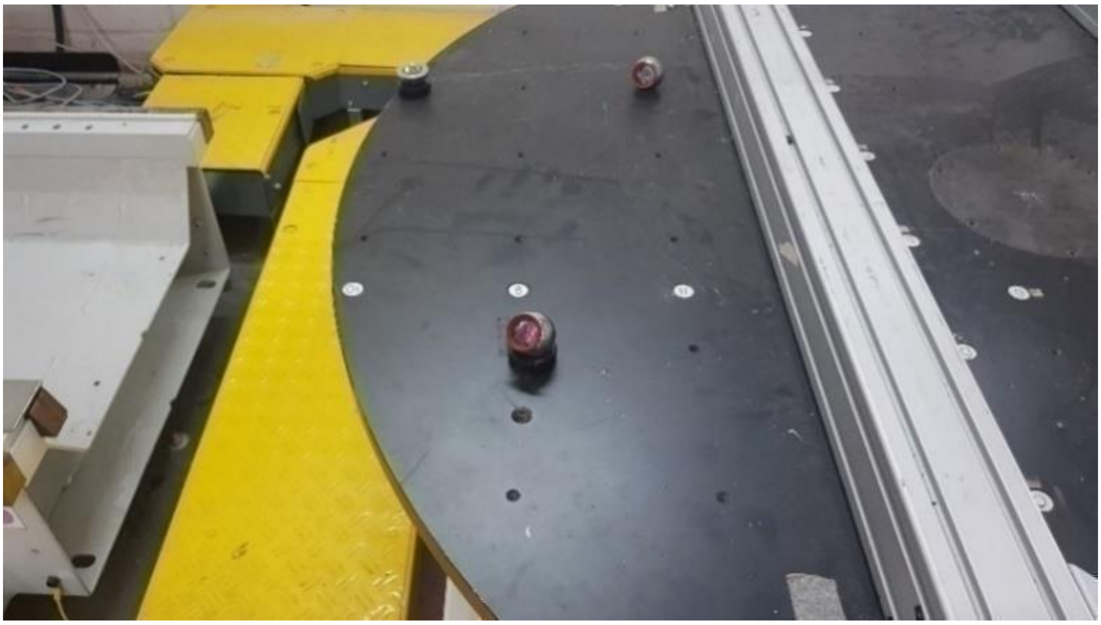


7.2.2.1 Turn table used as absolute workspace reference

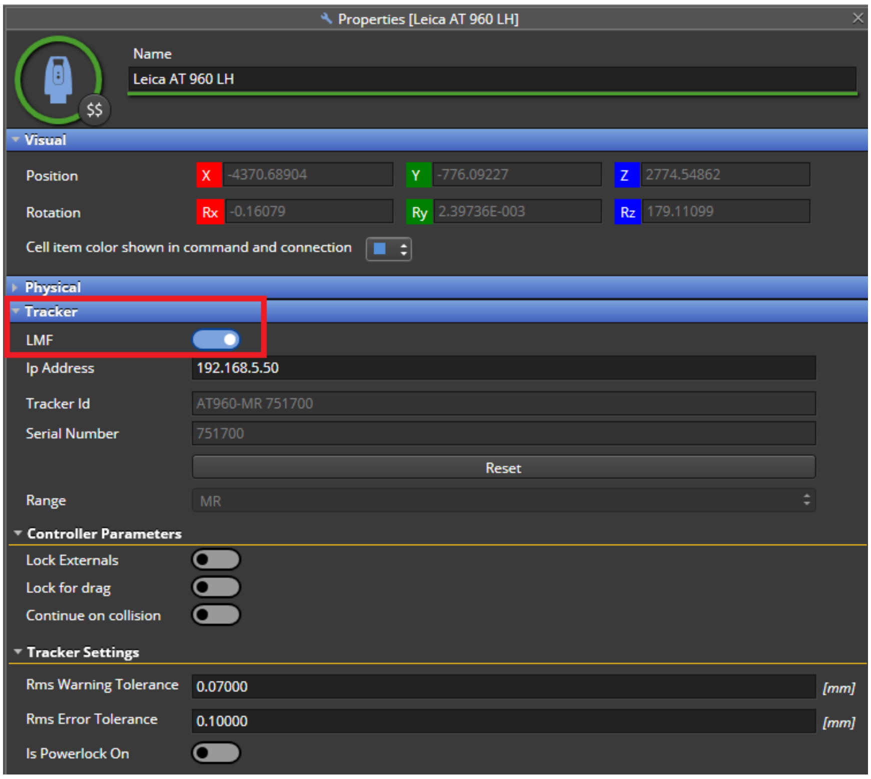
Turn Table alignment is done in the same as any object in the cell. It is possible to use the Cell alignment Wizard, or perform it manually by the following method that can be followed also to align any other static objects in the cell:

1) Use the Tracker in same position as it was in the Floor alignment as described in chapter 7.2.1.
2) Create three planes by positioning on each plane at least three Reflectors and measuring the points to create a plane.
3) Perform manual alignment, match the three planes in the CAD to the actual planes.

Place at least 3 Reflectors on the top plane of the actual Turn Table or you may use the same Reflector and measure and move it to create a plane on the top.



Connect to the Tracker by enabling the LMF toggle switch and measure the three reflectors. Chapter 7.2.1 can be taken as a reference since the same procedure to measure reflector positions applies here.



Properties [Leica AT 960 LH]

Name: Leica AT 960 LH

Visual

Position	X -4370.68904	Y -776.09227	Z 2774.54862
Rotation	Rx -0.16079	Ry 2.39736E-003	Rz 179.11099

Cell item color shown in command and connection: 🟦

Physical

Tracker

LMF	● on
Ip Address	192.168.5.50
Tracker Id	AT960-MR 751700
Serial Number	751700
	Reset
Range	MR

Controller Parameters

Lock Externals	○ off
Lock for drag	○ off
Continue on collision	○ off

Tracker Settings

Rms Warning Tolerance	0.07000	[mm]
Rms Error Tolerance	0.10000	[mm]
Is Powerlock On	○ off	

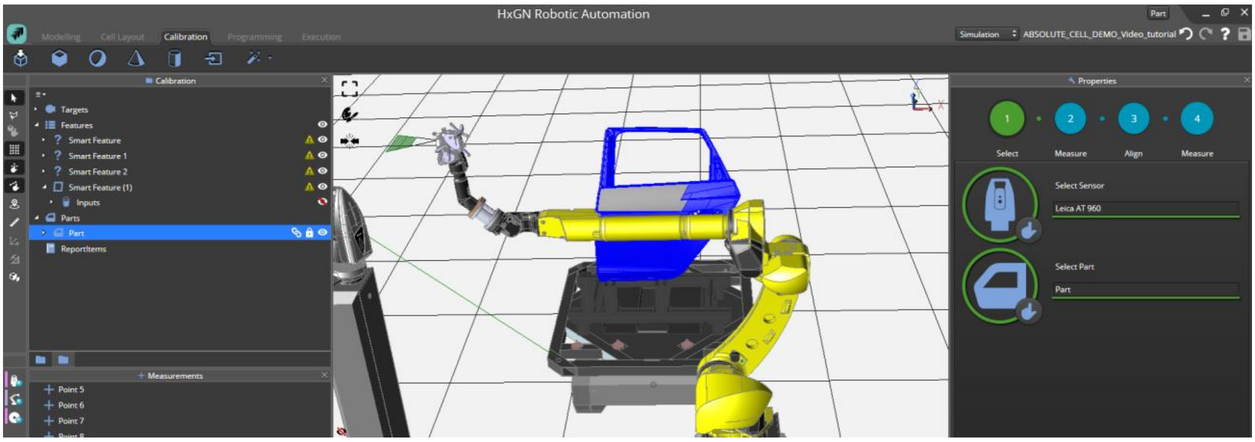
The user can continue to follow the remaining steps described in the cell alignment wizard to complete the alignment using the turn table as the absolute workspace reference.

7.2.3 Part alignment wizard

This wizard is used to align a part to its real cell position forcing the user through the following flow:

Select → Measure → Align → Measure

1) **Select** the part from 3D view or the project tree and follow the steps.



Select Sensor: Leica AT 960

Select Part: Part

2) **Measure** the feature positions needed to align the part, see similar step in chapter 7.2.2

3) **Align** the CAD to the measured positions, see also similar step chapter 7.2.2

The fourth step can be skipped and user can click **Finish** to complete the wizard.

7.2.4 Best fit alignment wizard

Warning: if the tracker is mounted on a slider (see chapter 5.1.2), the user must first unassign the tracker from the slider, align it, and then assign it again once aligned.

The goal is to align a new Laser Tracker in an existing project for the following three use cases:

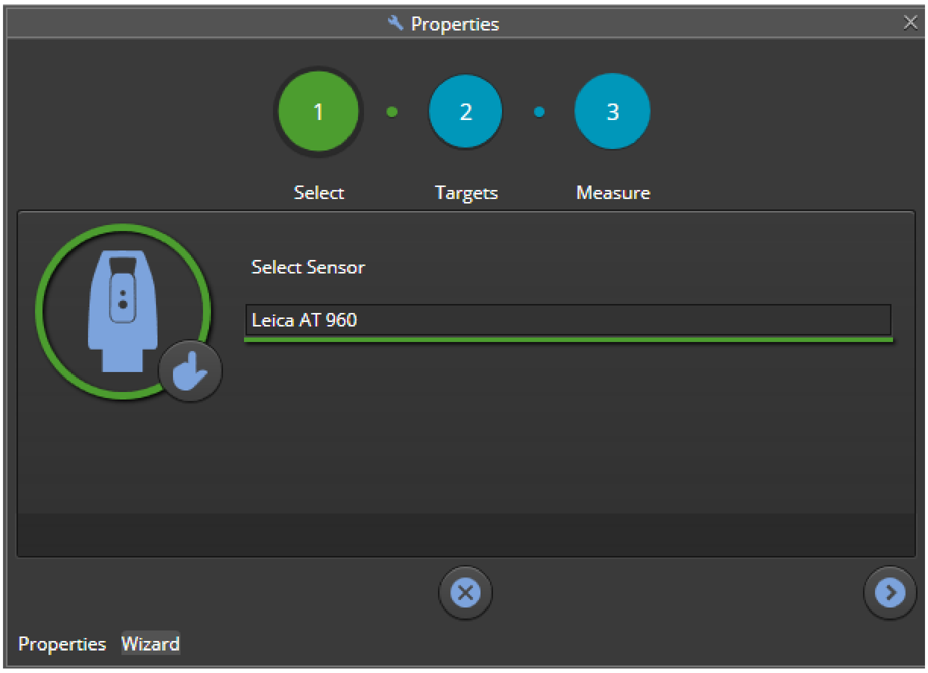
1) A new tracker is **added**.
2) The existing Tracker is **replaced** with a new one and the existing programs have to be updated.
3) The existing Tracker is **moved** to a new position and the existing programs have to be updated.

This alignment is usually applied after cell alignment was performed and measurement programs were generated. If for any reason it is necessary to move the Tracker to a new position, the goal of Best fit Alignment is to perform the alignment of the Tracker in the new position, using some reference points measured by the Tracker in the original position.

A preliminary requirement is to have a group at least 3 Targets (reflectors) measured with the Tracker in the original position.

At this point is possible to follow the Best Fit Alignment wizard steps:

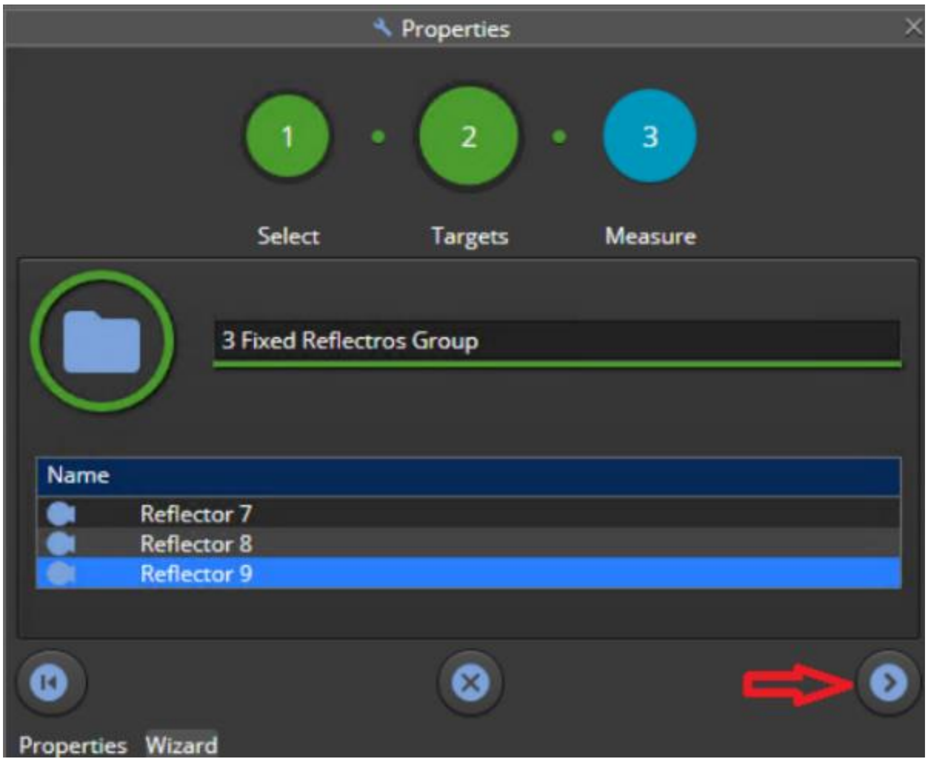
1) Select the Tracker (e.g. Leica AT960) and click Next.



Select Sensor

Leica AT 960

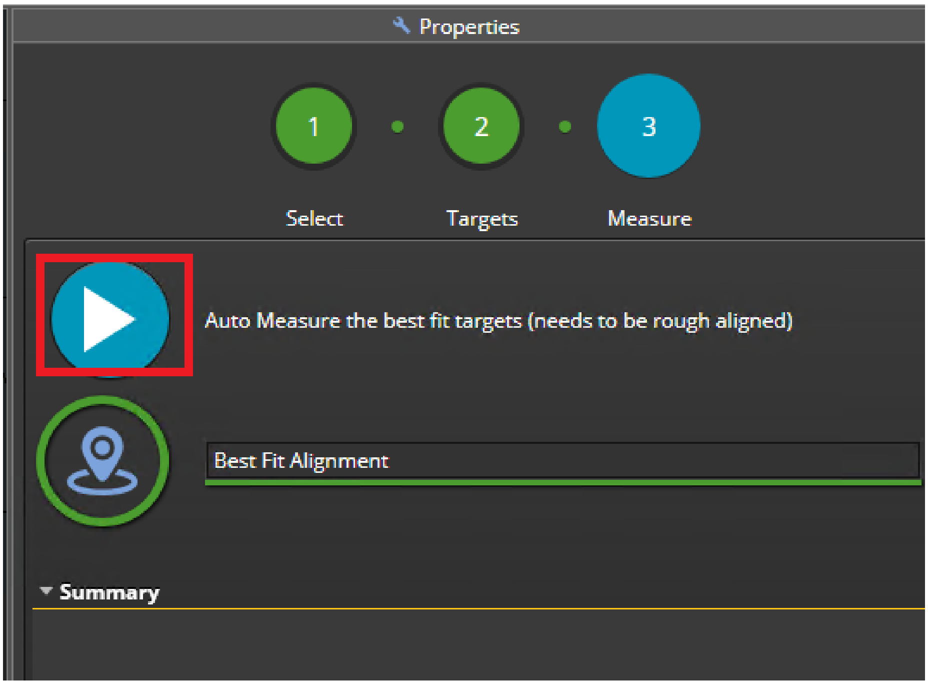
2) Select the existing target group in the project tree and the reflectors will appear into the Wizard. Click Next once the existing target group is selected and shown.



3 Fixed Reflectros Group

Name
Reflector 7
Reflector 8
Reflector 9

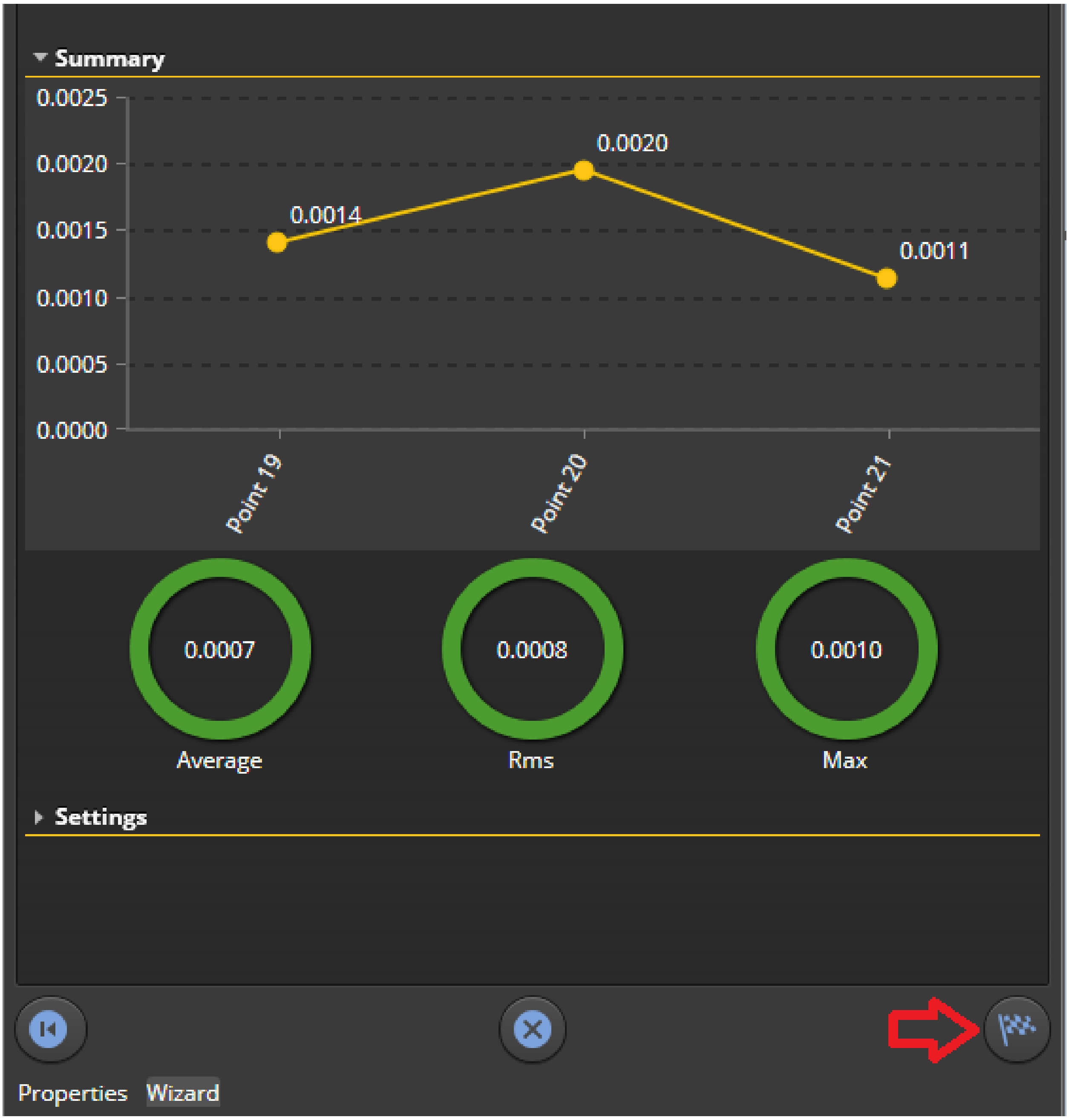
3) Click on Auto Measure and the system will automatically measure the targets with the new tracker. In case the automatic measure is not successful, it is also possible to measure manually the reflectors using the tracker teach pendant as explained in chapter 7.2.5



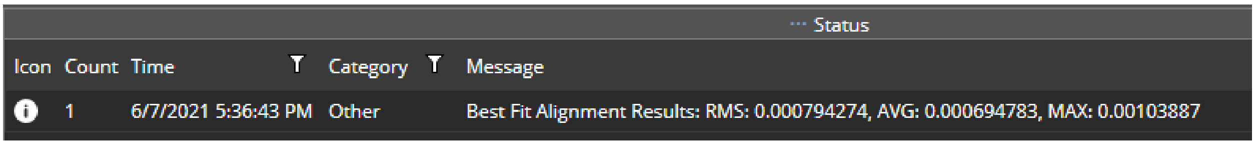
Auto Measure the best fit targets (needs to be rough aligned)

Best Fit Alignment

4) Click Finish once the summary is displayed.



5) You can also see the result in the status window.

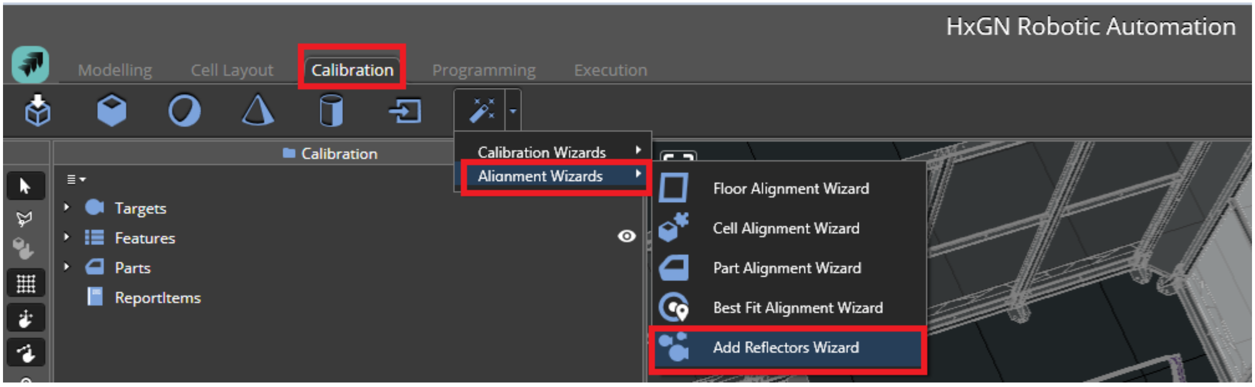


··· Status

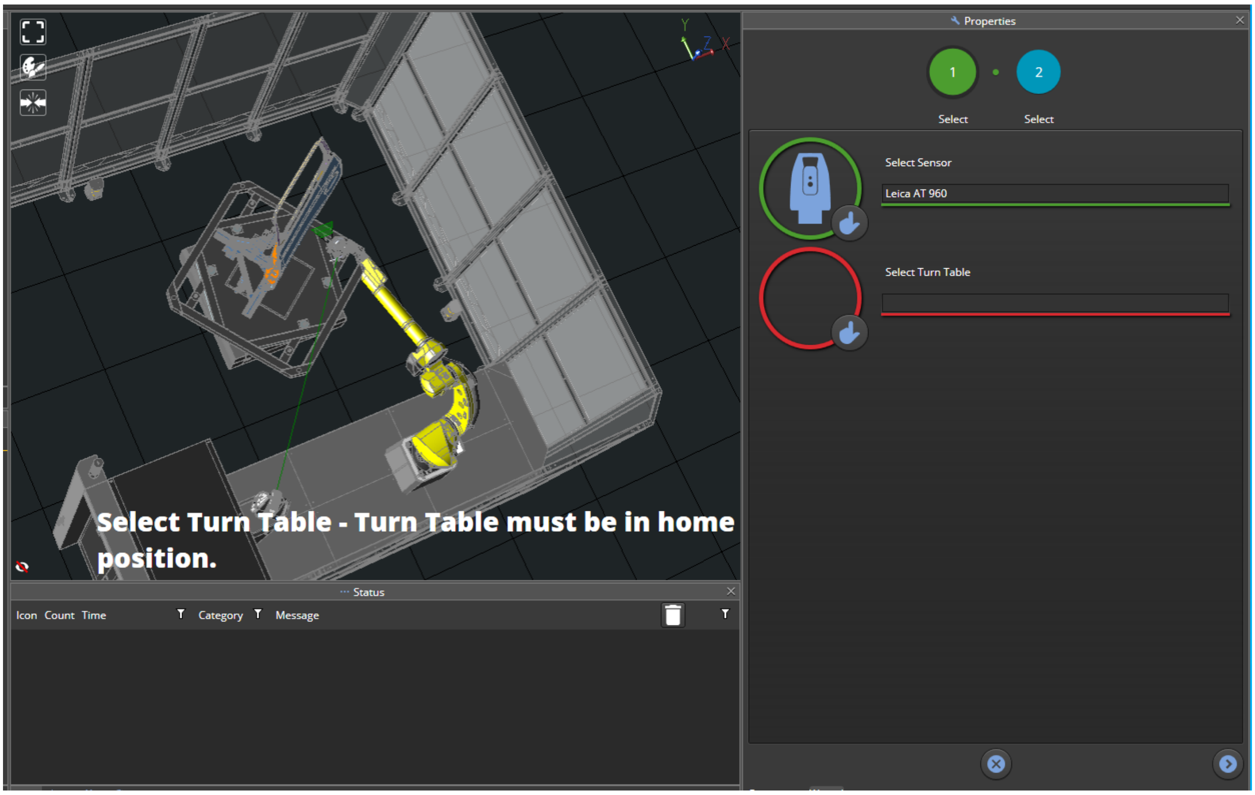
Icon	Count	Time	Category	Message
ⓘ	1	6/7/2021 5:36:43 PM	Other	Best Fit Alignment Results: RMS: 0.000794274, AVG: 0.000694783, MAX: 0.00103887

7.2.5 Add Reflectors Wizard

Under Calibration Tab, there is a wizard to help user insert Reflectors:

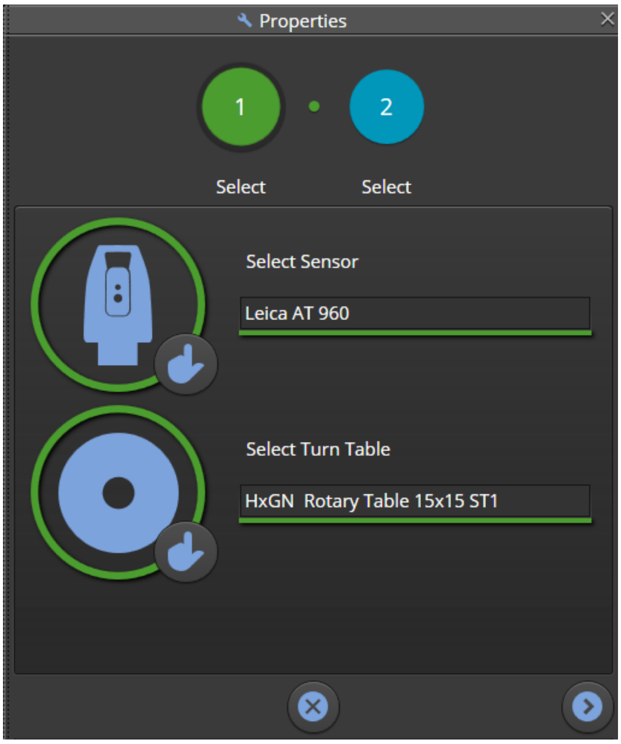


Once the wizard properties opens on the right, you may select Turn table if reflectors are on the Turn table (make sure the Turn table is in the home position): the reflectors will move together with the Turn table, or skip and click Next (button right): the reflectors will be static.


Select Turn Table - Turn Table must be in home position.

Select Sensor: Leica AT 960

Select Turn Table:



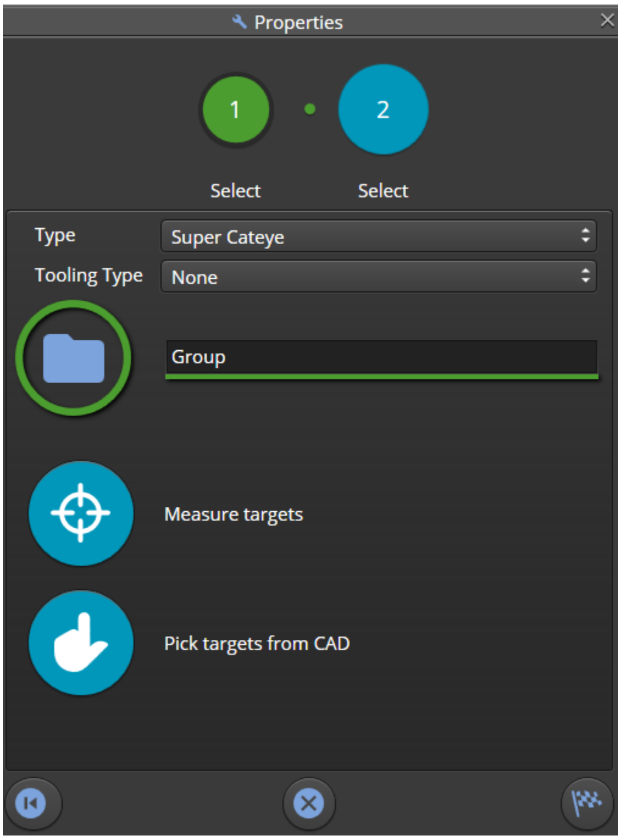
Select Sensor

Leica AT 960

Select Turn Table

HxGN Rotary Table 15x15 ST1

Next step is to select the reflector type, its tooling type, and it is also possible to rename the group as preferred in case multiple groups are present in the project:



Type: Super Cateye

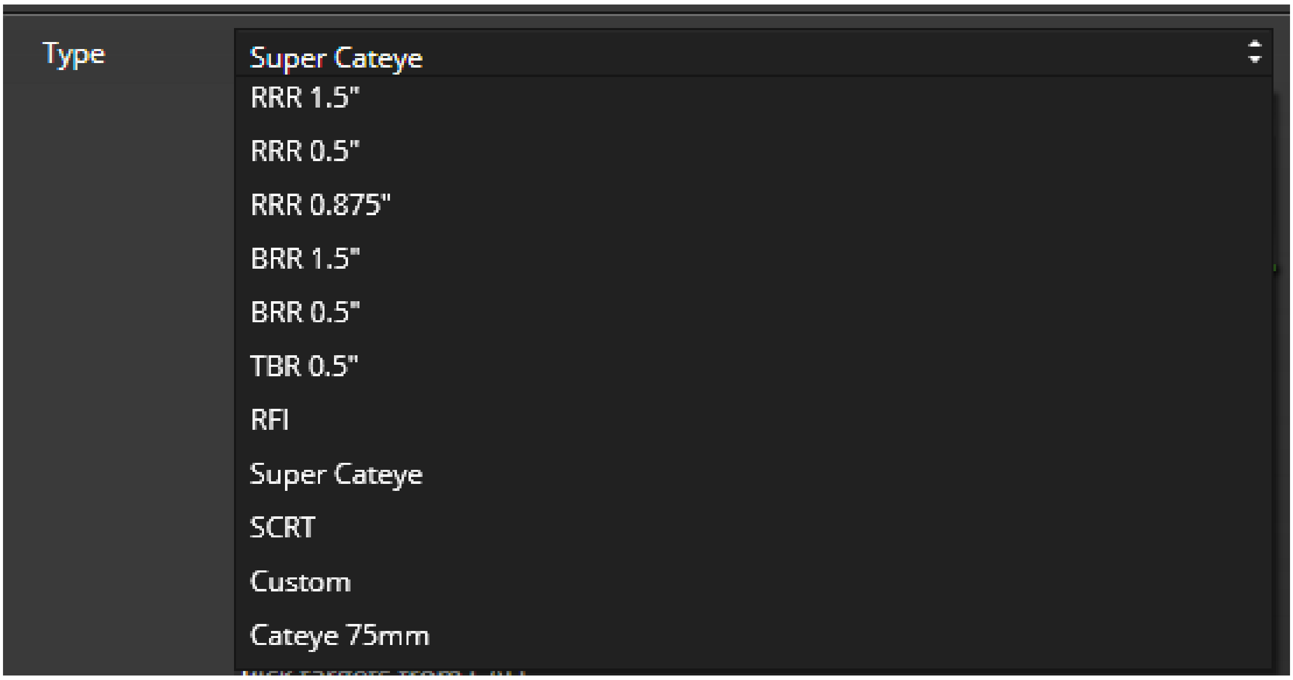
Tooling Type: None

Group

Measure targets

Pick targets from CAD

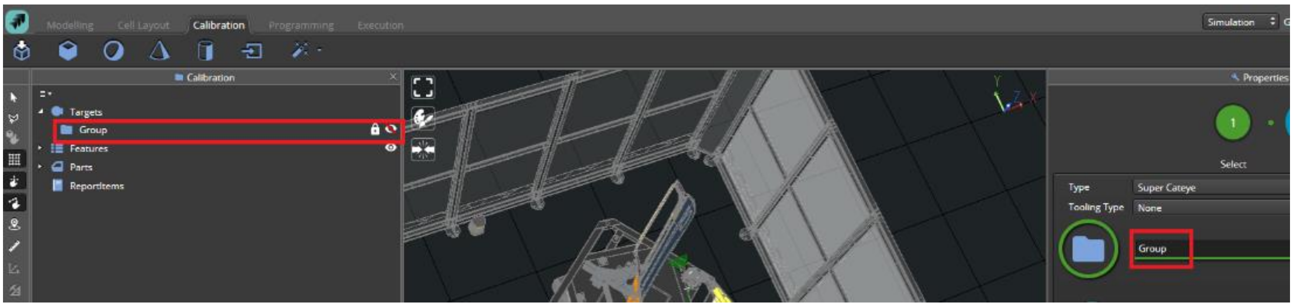
Select the reflector type from the list of supported reflector types in the drop-down list:



Type: Super Cateye

- RRR 1.5"
- RRR 0.5"
- RRR 0.875"
- BRR 1.5"
- BRR 0.5"
- TBR 0.5"
- RFI
- Super Cateye
- SCRT
- Custom
- Cateye 75mm

Select the Tooling Type and define a name for the Group:

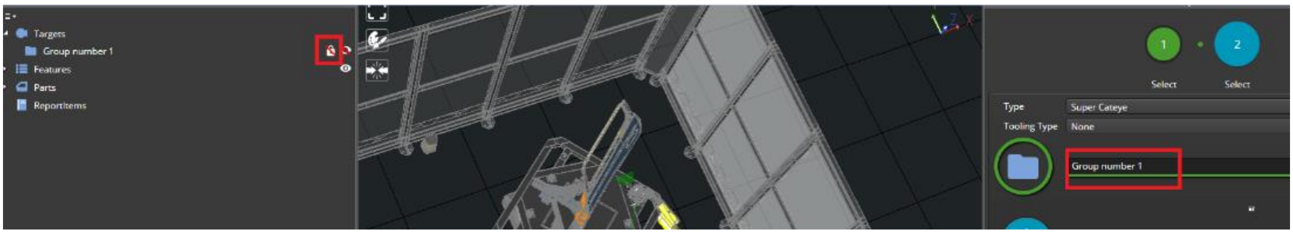


Type: Super Cateye

Tooling Type: None

Group

To change group name, it is needed to unlock the target group name from the project tree:



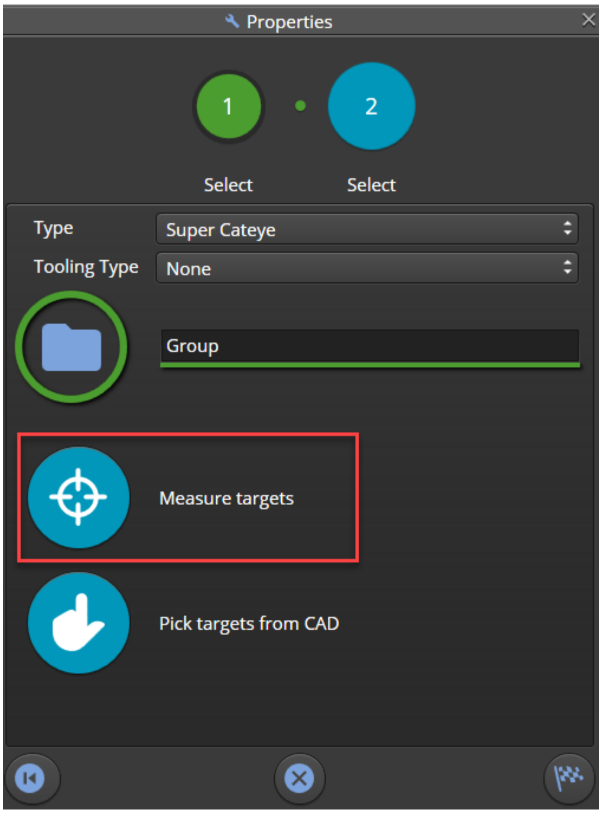
Type: Super Cateye

Tooling Type: None

Group number 1

It is now possible to choose either to:

1) **Measure targets**: for measuring existing targets in the real cell by using the tracker teach pendant as already shown in chapter 7.1.1

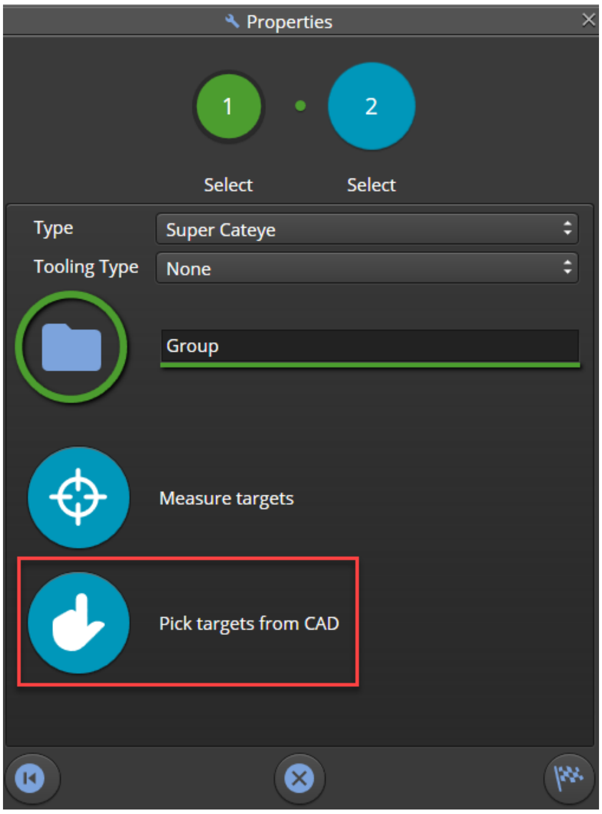


Properties

Type	Super Cateye
Tooling Type	None

Group

2) **Pick targets from CAD**: for picking the position of reflectors from the 3D View (used for offline mode).

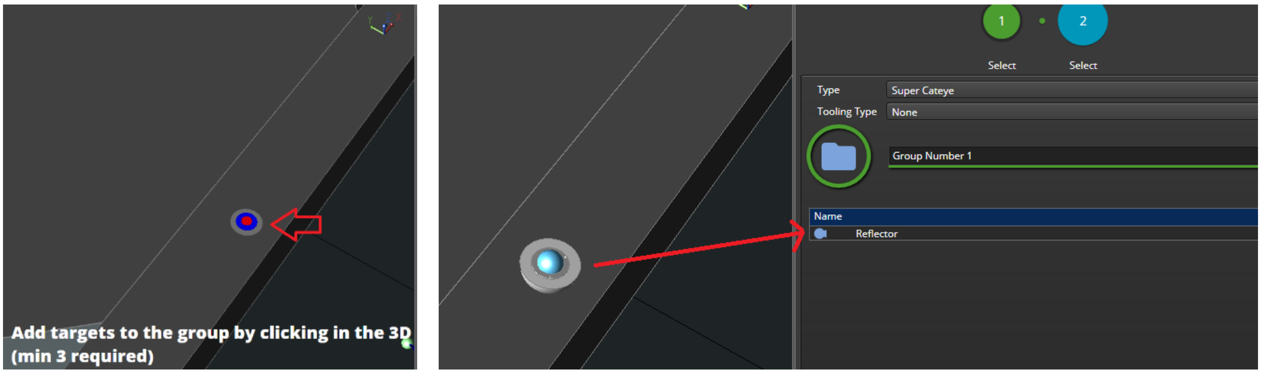


Properties

Type	Super Cateye
Tooling Type	None

Group

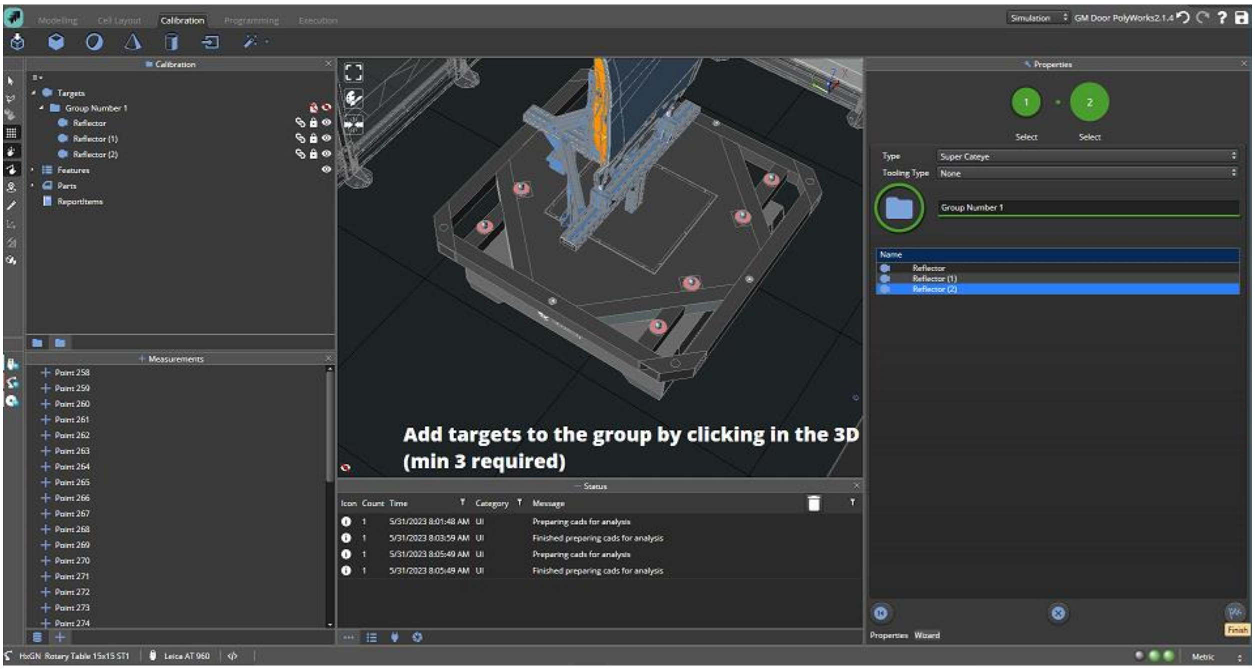
By clicking with the mouse, the reflector will be added within the group:


Add targets to the group by clicking in the 3D (min 3 required)

Type: Super Cateye
Tooling Type: None
Group Number 1

Name
Reflector

Continue with at least 3 reflectors. Once all reflectors are added, click **Finish** to complete the wizard.


Add targets to the group by clicking in the 3D (min 3 required)

Type: Super Cateye
Tooling Type: None
Group Number 1

Name
Reflector
Reflector (1)
Reflector (2)

Status

Icon	Count	Time	Category	Message
ⓘ	1	5/31/2023 8:01:48 AM	UI	Preparing cads for analysis
ⓘ	1	5/31/2023 8:03:59 AM	UI	Finished preparing cads for analysis
ⓘ	1	5/31/2023 8:05:49 AM	UI	Preparing cads for analysis
ⓘ	1	5/31/2023 8:05:49 AM	UI	Finished preparing cads for analysis

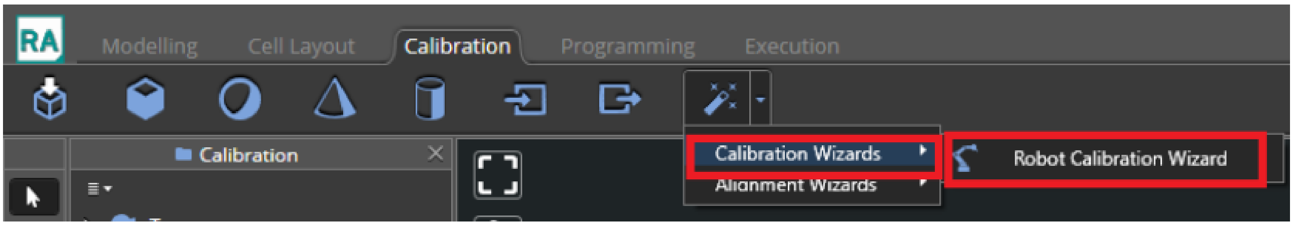
7.3 Dual tracker cell alignment workflow

In cells where more than one tracker is available to measure, it is necessary as a first step to align them each other.
In this way it does not matter which tracker is used to measure the features as all of them will be referred to the same coordinate system. The alignment process for a single tracker is described in chapter 7.2.4.
Once both trackers are aligned, the user needs to follow the workflow described below:

1) Measure all possible features with the main tracker
2) Execute the cell alignment wizard following all the steps
3) Align the secondary tracker using the best fit alignment wizard
4) Measure with the secondary tracker all the features that could not be measured by the main tracker
5) Use the alignment mode to align the new features measure in the previous step
6) In case the tracker is mounted on a slider, it needs to be linked to the respective slider
7) Be sure that all the features are locked in the project tree to avoid any unwanted movements

7.4 Robot alignment

The Robot alignment process must be done to adjust the robot theoretical model to the real robot used in the cell and to align the robot base with the tool TCP.



A basic alignment is done using the Robot Calibration wizard for the following:

- Robot base – 6 DOF (degree of Freedom) position. In case the Robot in mounted on a slider, the slider 6DOF position must be considered.
- Tool TCP (T-Scan or AS1 laser scanners) – 6DOF position. With a basic alignment, it is expected to get good accuracy results of RMS below 2 mm.

This robot alignment is not focused on advanced calibration of robot parameters (e.g. motor encoder calibration) that must be executed according to manufacturers guidelines.

Perquisites:

1) Leica Metrology Foundation (LMF) and Tracker installed on the system (AT 960-MR)
2) LMF is the only tool that can be used for robot alignment (not possible using T-Collect or RDS)
3) Cell model imported to the online HxGN Robotic Automation application.
4) Cell.json file modified to have a workstation configured for robot calibration purposes (meaning that the tracker must not be included within the workstation as it must be controlled by LMF and not by HxGN Robotic Automation) with unique ID (e.g. 100). This workstation ID must be used for the robot calibration process as only the correct cell elements will be used.
5) Imported Cell is already aligned (all elements) as described in previous chapters.
6) Only a single alignment per robot is required. In case a dual robot cell is used, every robot must be aligned separately.

The alignment is done in the context of the workspace you intend to use, i.e. the area where the part is placed to be measured. E.g., the workspace can be the area above the turn table when it is used as a part holder where to place the part and the relative fixture.

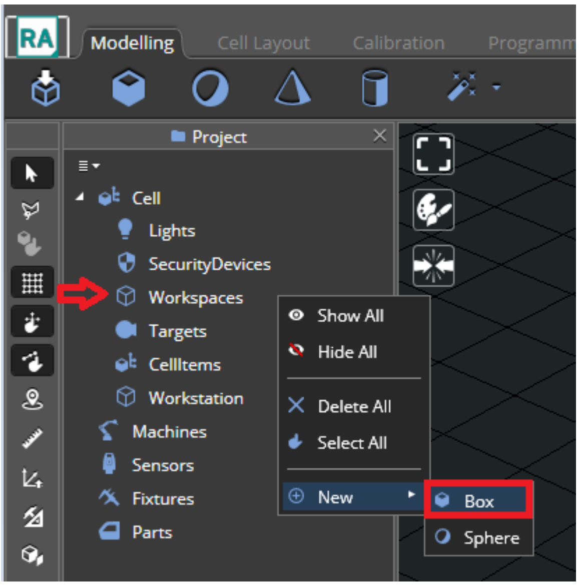
In a cell where the robot is used in two distinct workspaces, e.g., two turn tables, it is recommended to do two alignments in two different projects.

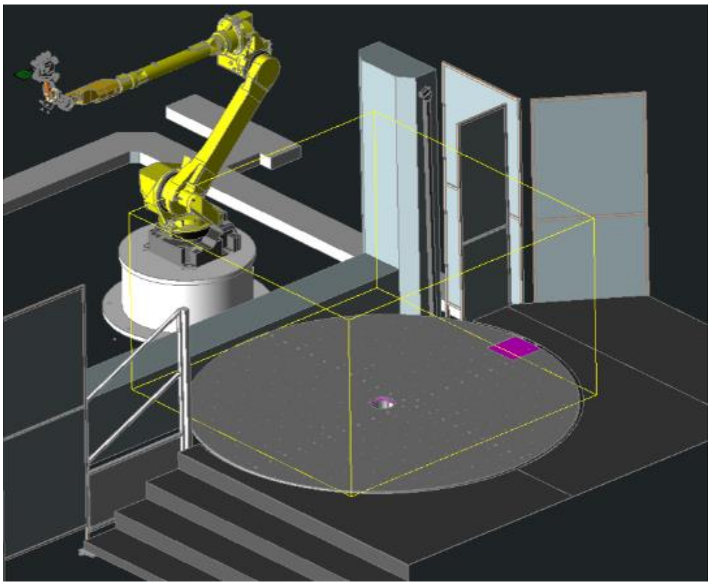
To create a workspace, in Modelling mode, select a new **Box** workspace and adjust the workspace size and position to reflect your working area. This is the area which your largest part to be measured will occupy (yellow wireframe box). Make sure to define workspace before you start the wizard.

Pro Tip: users need to take into account that:

- the minimum distance between tracker and scanner is 1.5 m
- the solution provided by the path planning algorithm will always try to place the scanner parallel to the workspace face and with a face pointed towards the tracker in order for it to lock on that face of the scanner.

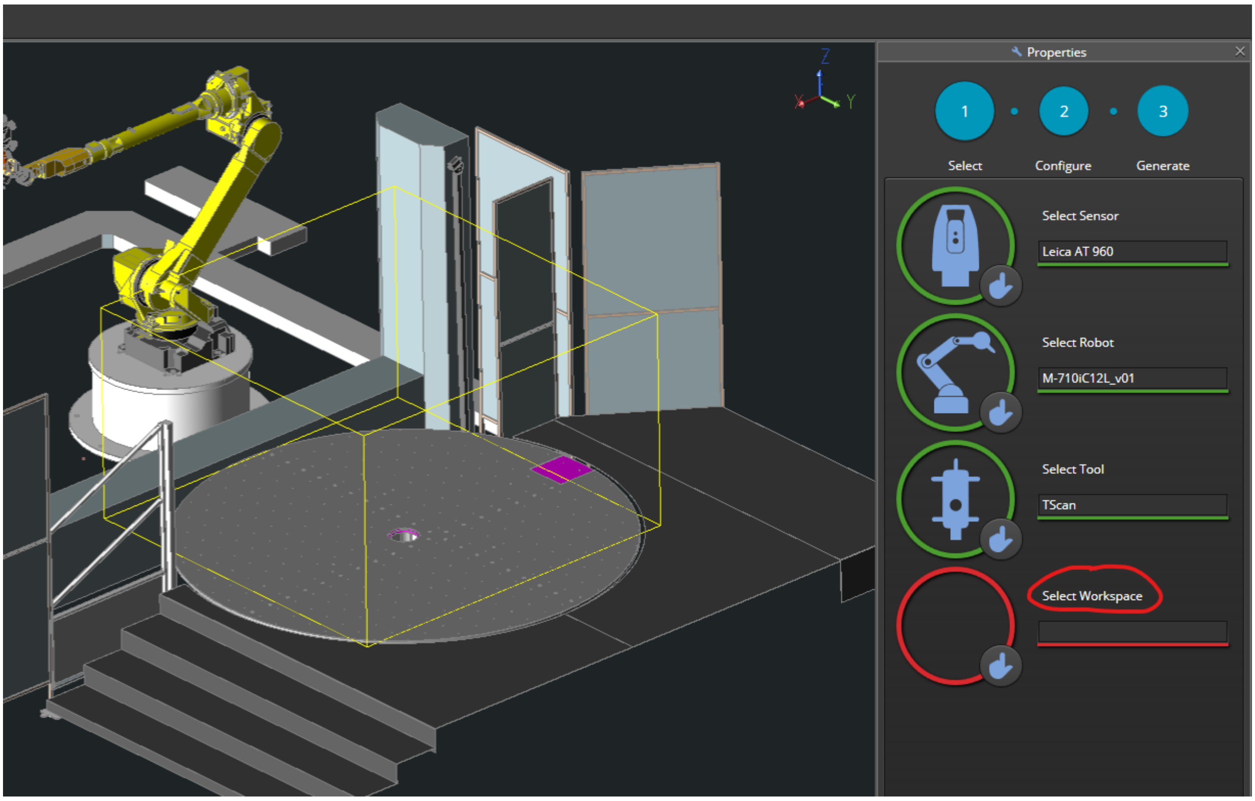
If the conditions above are not taken into account, it may be that the face on which the tracker needs to lock will not be visible and the algorithm to fail.





The wizard can be executed, and the steps followed:

1) Select your workspace and press Next.



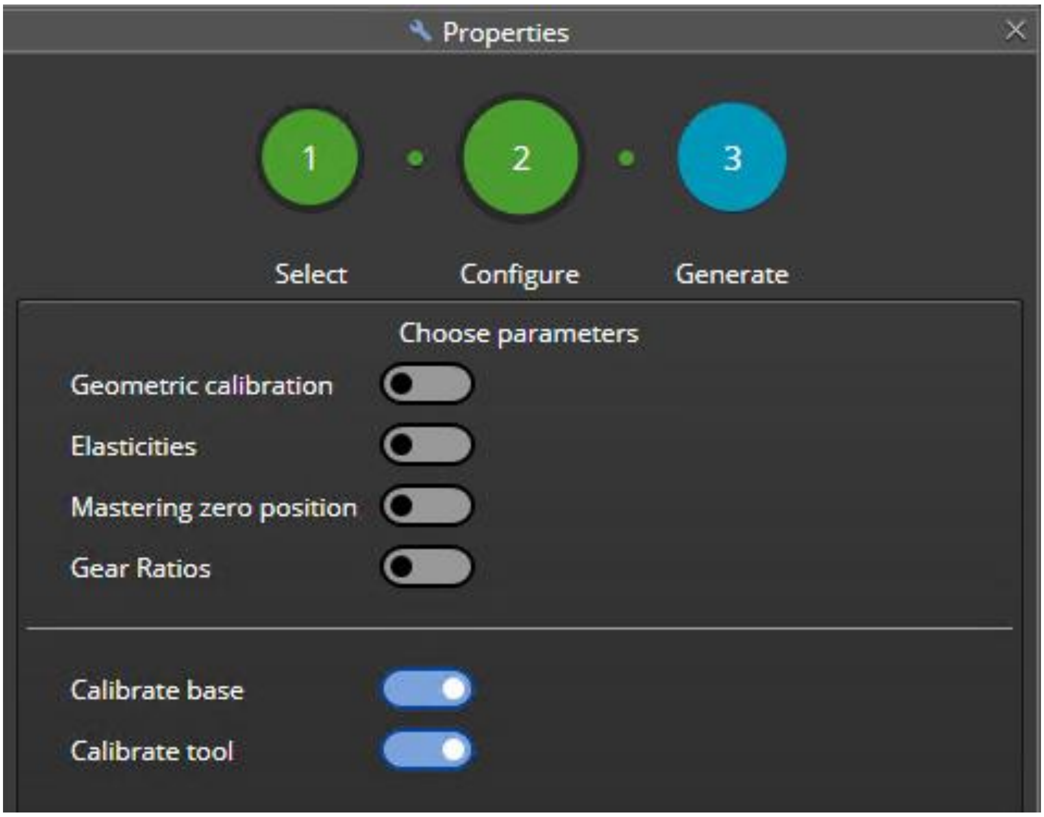
Select Sensor: Leica AT 960

Select Robot: M-710iC12L_v01

Select Tool: TScan

Select Workspace:

2) Set the parameters as default values shown in the following picture and continue to the next step.



Choose parameters

Geometric calibration ○

Elasticities ○

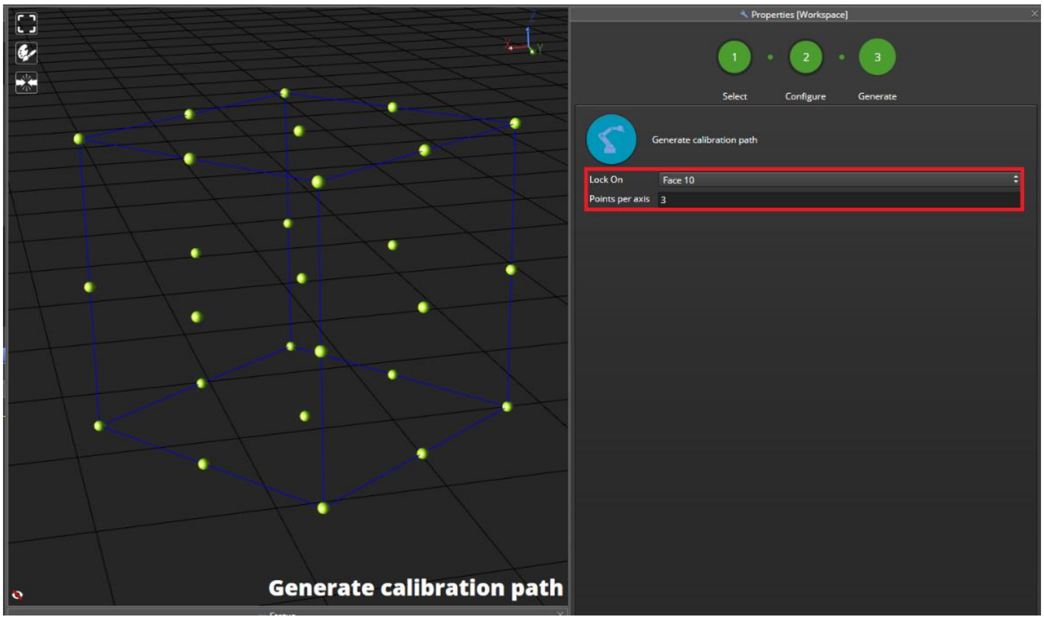
Mastering zero position ○

Gear Ratios ○

Calibrate base ●

Calibrate tool ●

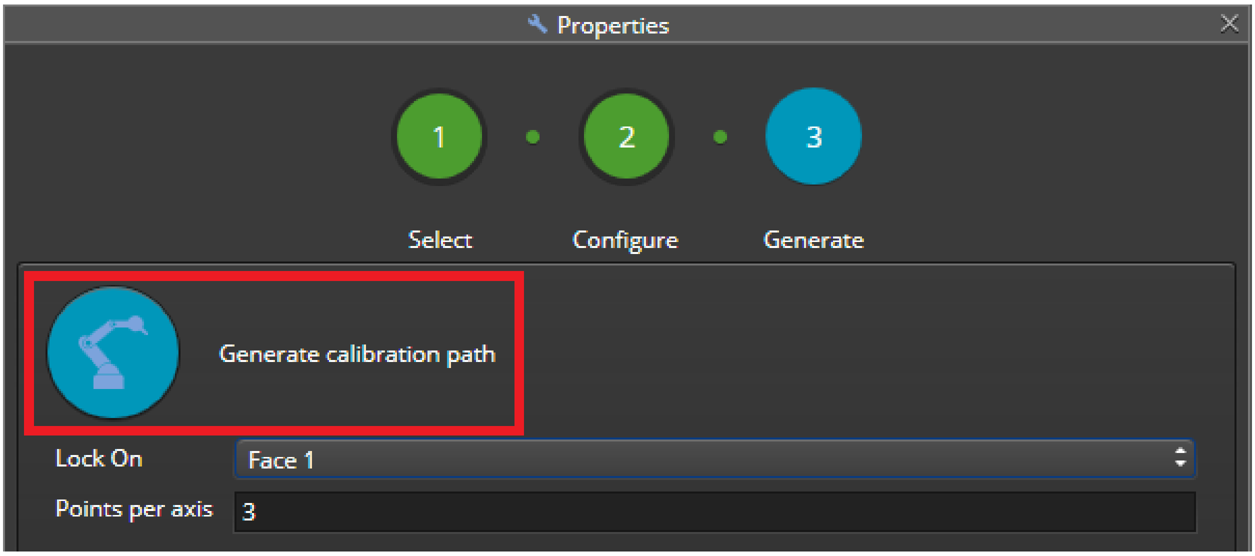
3) In this step, we will generate the robot alignment program by setting two parameters:
   a) **Lock On**: the scanner face to be used to lock on it. It is recommended to use first one side face of the scanner for the most stable results. It is usually suggested to use the left face of the scanner (Face 1)
   b) **Points per axis**: the number of points which will be used for alignment. This number decides the discretization of the workspace. It is suggested to use the default value (3). It will create a grid of 3x3x3 (total of 27 points per face). This is good to have redundancy as many points will not be reachable.



Lock On: Face 10

Points per axis: 3

4) Click on “Generate calibration path” and the program will be automatically created. Click on Finish icon to complete the wizard.



Lock On: Face 1

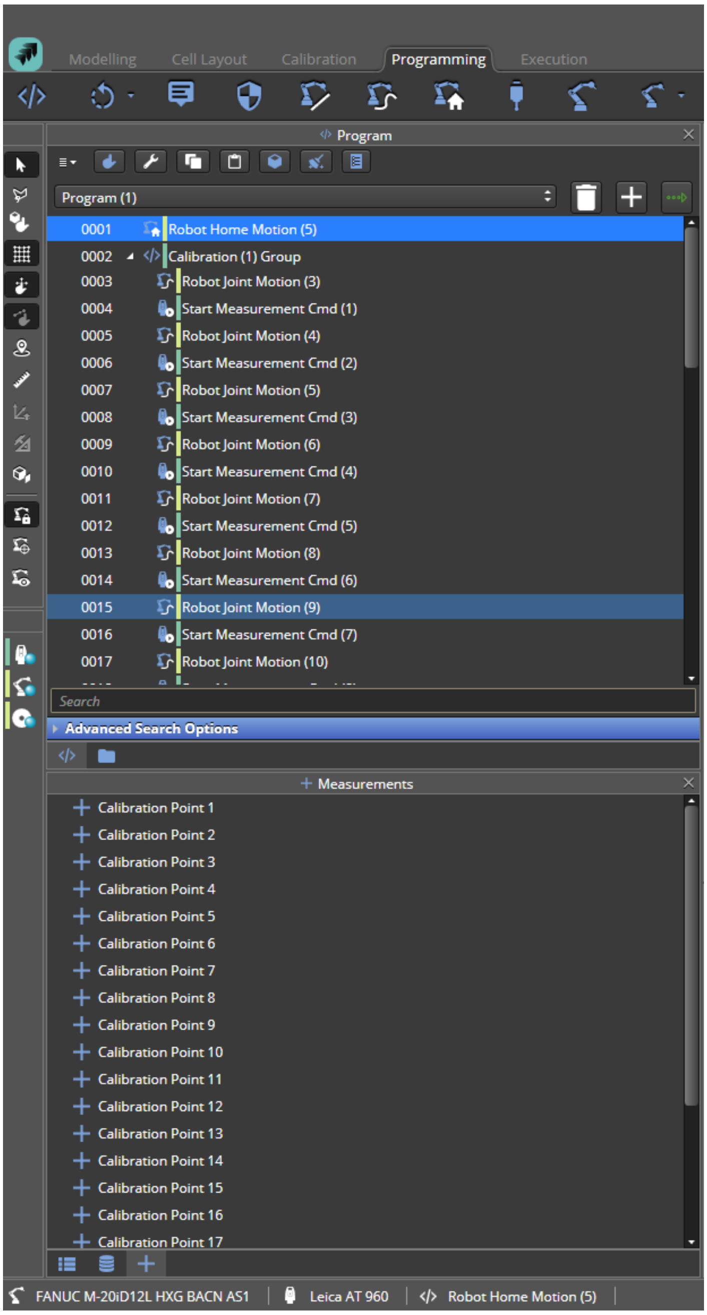
Points per axis: 3

The calibration wizard generated a program for the face we chose (Face 1).

5) User now needs to validate the program and execute it in the real cell. After the execution is completed, the calibration points will be available in the project tree under **Calibrations** inside the robot that was used in the calibration wizard.

Warning: to validate the program and run it an online environment within HxGN Robotic Automation Execution mode, it is strongly suggested to use slow speed.

Info: make sure there are no Calibration Points in the project before executing the program in online mode otherwise the results may be wrong.

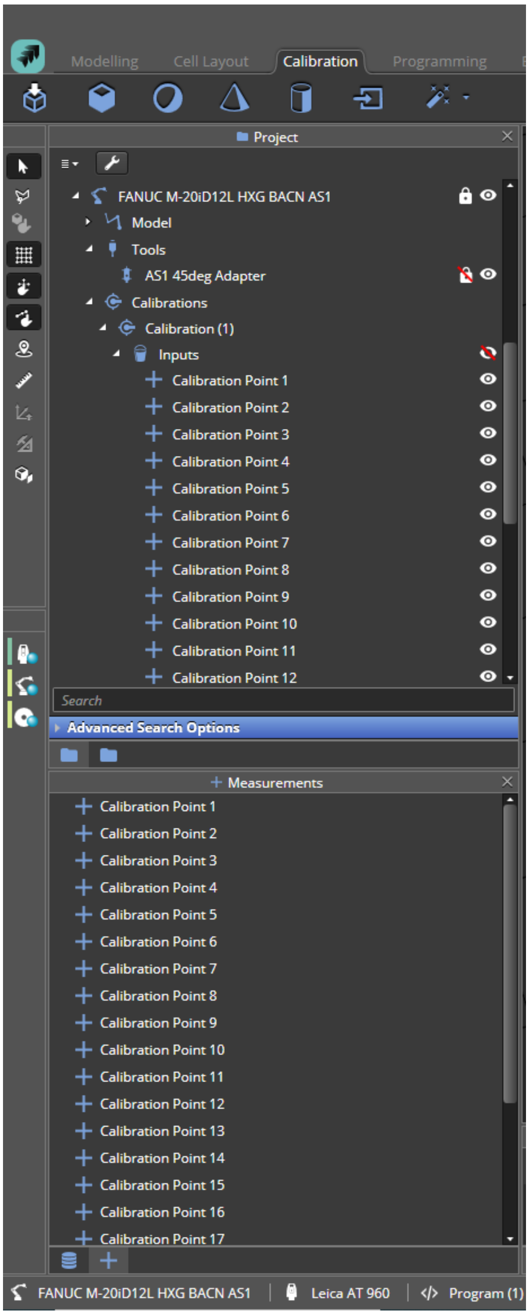


Program (1)

0001 Robot Home Motion (5)
0002 Calibration (1) Group
0003 Robot Joint Motion (3)
0004 Start Measurement Cmd (1)
0005 Robot Joint Motion (4)
0006 Start Measurement Cmd (2)
0007 Robot Joint Motion (5)
0008 Start Measurement Cmd (3)
0009 Robot Joint Motion (6)
0010 Start Measurement Cmd (4)
0011 Robot Joint Motion (7)
0012 Start Measurement Cmd (5)
0013 Robot Joint Motion (8)
0014 Start Measurement Cmd (6)
0015 Robot Joint Motion (9)
0016 Start Measurement Cmd (7)
0017 Robot Joint Motion (10)

Calibration Point 1
Calibration Point 2
Calibration Point 3
Calibration Point 4
Calibration Point 5
Calibration Point 6
Calibration Point 7
Calibration Point 8
Calibration Point 9
Calibration Point 10
Calibration Point 11
Calibration Point 12
Calibration Point 13
Calibration Point 14
Calibration Point 15
Calibration Point 16
Calibration Point 17

FANUC M-20iD12L HXG BACN AS1 | Leica AT 960 | Robot Home Motion (5)



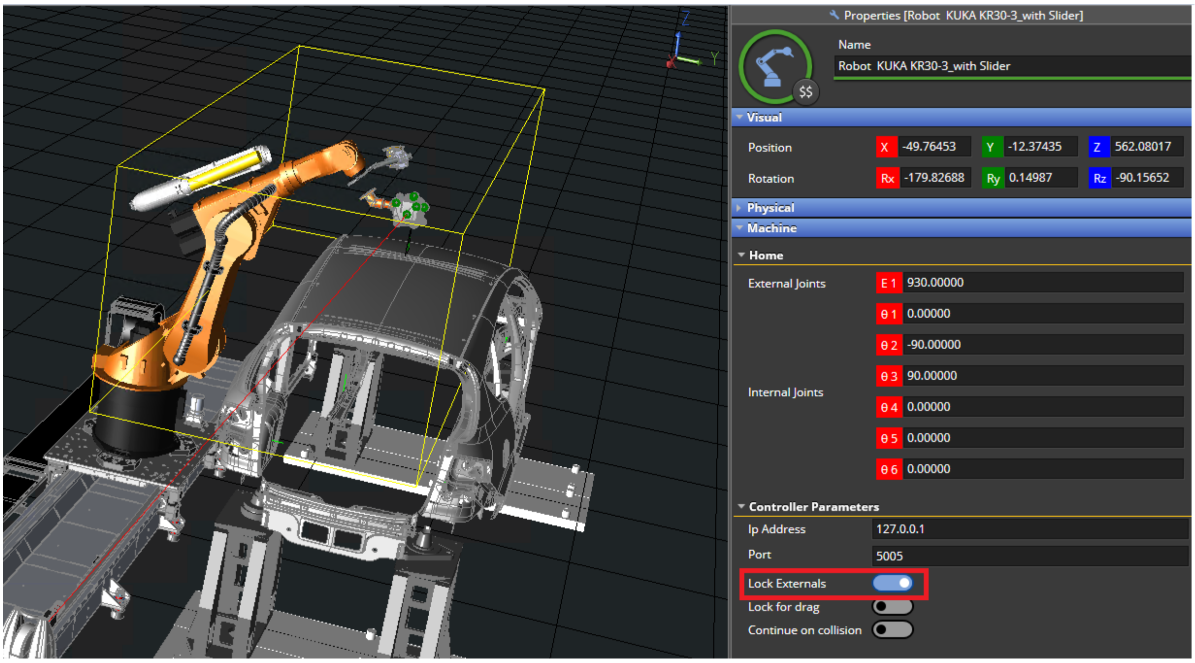
FANUC M-20iD12L HXG BACN AS1
Model
Tools
AS1 45deg Adapter
Calibrations
Calibration (1)
Inputs
Calibration Point 1
Calibration Point 2
Calibration Point 3
Calibration Point 4
Calibration Point 5
Calibration Point 6
Calibration Point 7
Calibration Point 8
Calibration Point 9
Calibration Point 10
Calibration Point 11
Calibration Point 12

Calibration Point 1
Calibration Point 2
Calibration Point 3
Calibration Point 4
Calibration Point 5
Calibration Point 6
Calibration Point 7
Calibration Point 8
Calibration Point 9
Calibration Point 10
Calibration Point 11
Calibration Point 12
Calibration Point 13
Calibration Point 14
Calibration Point 15
Calibration Point 16
Calibration Point 17

FANUC M-20iD12L HXG BACN AS1 | Leica AT 960 | Program (1)

It is needed to measure at least ten points in the workspace. It is also needed to get a variety of robot orientations, configurations, and joint values. In case of a slider is used, one slider position is to be used making sure to create a workspace box big enough to cover the part to be measured.

Robot alignment supports both 6 or 7 axes configurations. If the robot is moving on a slider (7 axes configuration) but when it reaches the workspace, the slider is not used to perform the scanning, there is no need to do the robot alignment considering that axis. For this use case, move the slider to the position where you want to perform the alignment on the workspace, enable **Lock Externals** toggle switch and use the robot calibration wizard.



Properties [Robot KUKA KR30-3_with Slider]

Name: Robot KUKA KR30-3_with Slider

Visual

Position	X	-49.76453	Y	-12.37435	Z	562.08017
Rotation	Rx	-179.82688	Ry	0.14987	Rz	-90.15652

Physical

Machine

Home

External Joints	E 1	930.00000
Internal Joints	θ 1	0.00000
	θ 2	-90.00000
	θ 3	90.00000
	θ 4	0.00000
	θ 5	0.00000
	θ 6	0.00000

Controller Parameters

Ip Address: 127.0.0.1

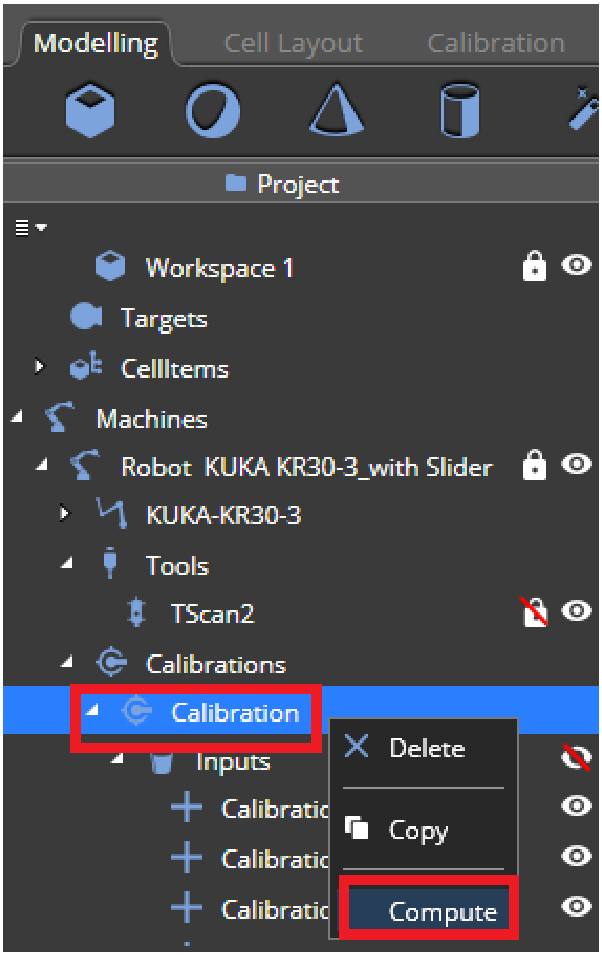
Port: 5005

Lock Externals: ●

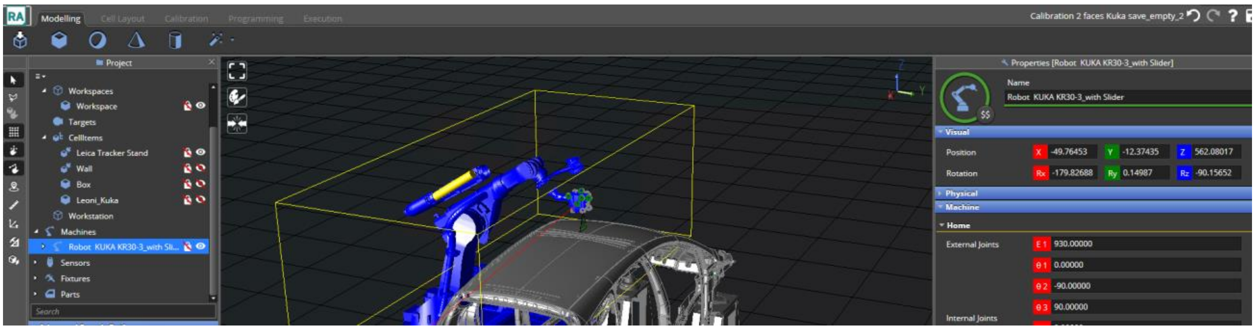
Lock for drag: ○

Continue on collision: ○

6) To execute the calibration, right click on *Calibration* | *Compute.*



The alignment will update the position of the robot and the Tool.



Name: Robot KUKA KR30-3_with Slider

Visual

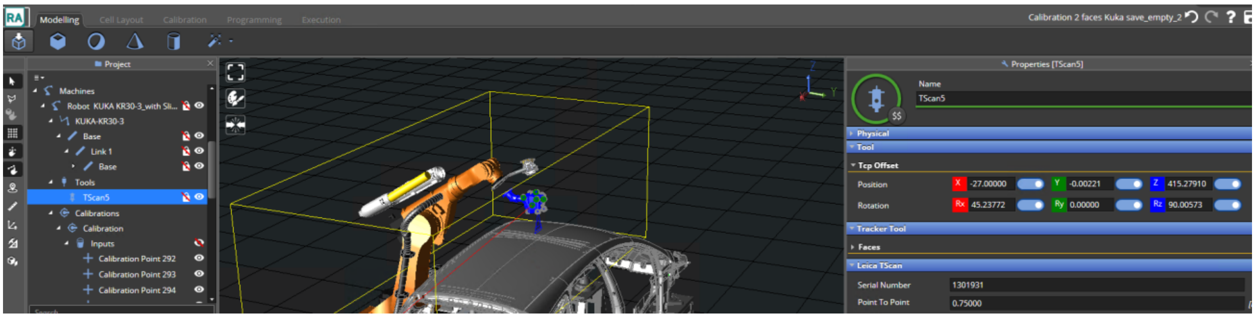
Position	X -49.76453	Y -12.37435	Z 562.08017
Rotation	Rx -179.82688	Ry 0.14987	Rz -90.15652

Physical

Machine

Home

External Joints	E 1 930.00000
	θ 1 0.00000
	θ 2 -90.00000
	θ 3 90.00000
Internal Joints	



Name: TScan5

Physical

Tool

Tcp Offset

Position	X -27.00000	Y -0.00221	Z 415.27910
Rotation	Rx 45.23772	Ry 0.00000	Rz 90.00573

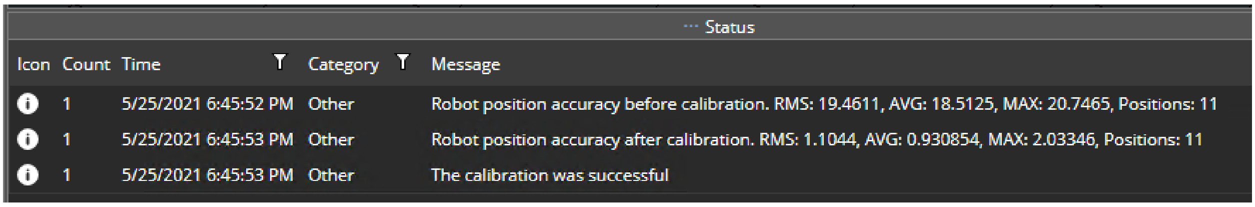
Tracker Tool

Faces

Leica TScan

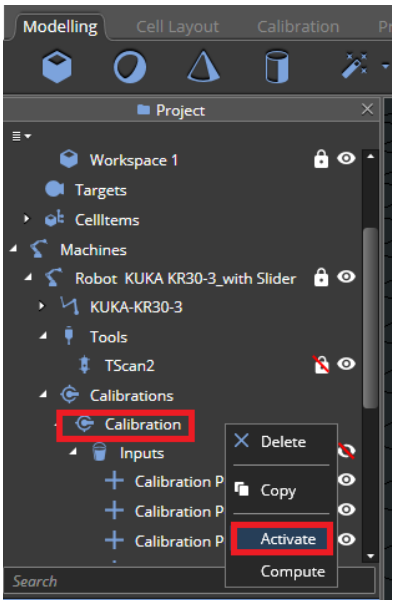
Serial Number	1301931
Point To Point	0.75000

The computation results will be displayed in the Status window. The alignment can be considered acceptable if the RMS is under 2 mm.



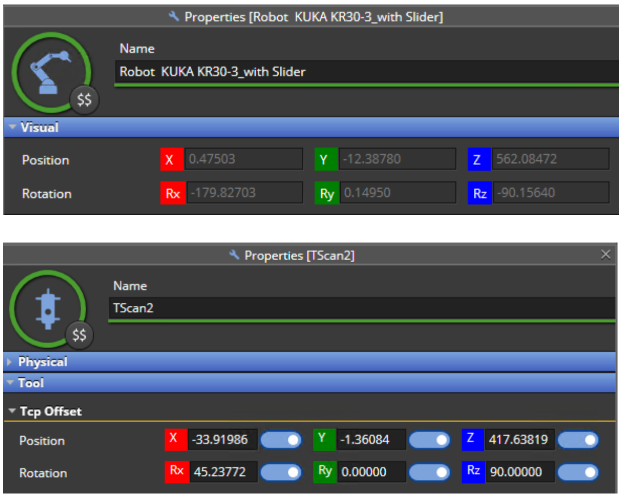
Icon	Count	Time	Category	Message
ⓘ	1	5/25/2021 6:45:52 PM	Other	Robot position accuracy before calibration. RMS: 19.4611, AVG: 18.5125, MAX: 20.7465, Positions: 11
ⓘ	1	5/25/2021 6:45:53 PM	Other	Robot position accuracy after calibration. RMS: 1.1044, AVG: 0.930854, MAX: 2.03346, Positions: 11
ⓘ	1	5/25/2021 6:45:53 PM	Other	The calibration was successful

Next, activate the calibration, by right-clicking on *Calibration* | *Activate*.



Note that the robot base changed slightly when the calibration was activated.

Before



Properties [Robot KUKA KR30-3_with Slider]

Name: Robot KUKA KR30-3_with Slider

Visual

	X	Y	Z
Position	0.47503	-12.38780	562.08472

	Rx	Ry	Rz
Rotation	-179.82703	0.14950	-90.15640

Properties [TScan2]

Name: TScan2

Physical

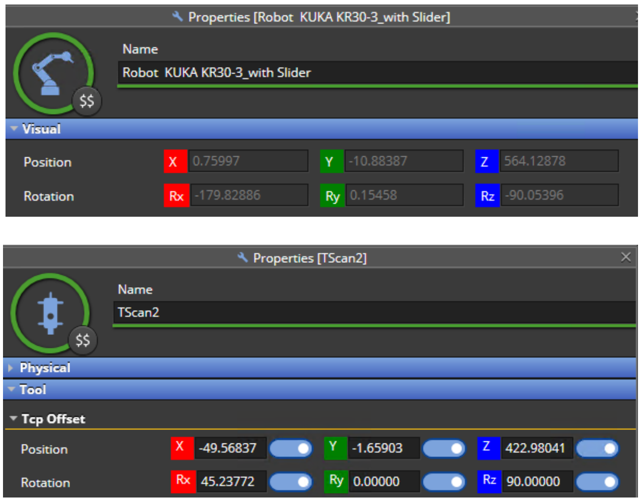
Tool

Tcp Offset

	X	Y	Z
Position	-33.91986	-1.36084	417.63819

	Rx	Ry	Rz
Rotation	45.23772	0.00000	90.00000

After



Properties [Robot KUKA KR30-3_with Slider]

Name: Robot KUKA KR30-3_with Slider

Visual

	X	Y	Z
Position	0.75997	-10.88387	564.12878

	Rx	Ry	Rz
Rotation	-179.82886	0.15458	-90.05396

Properties [TScan2]

Name: TScan2

Physical

Tool

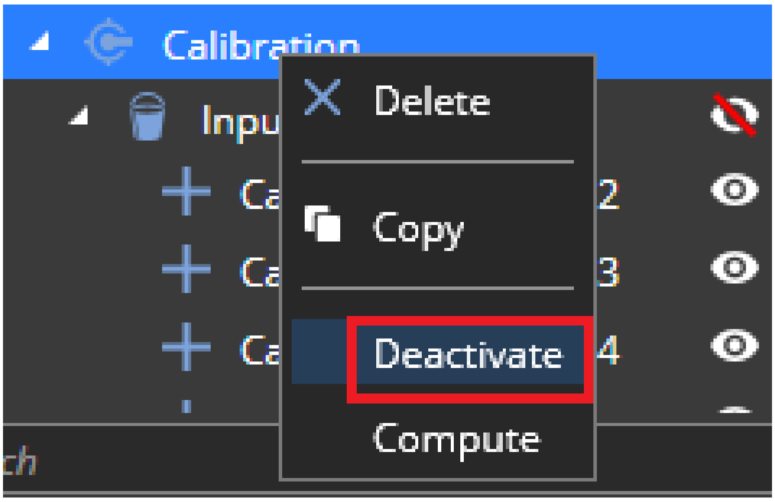
Tcp Offset

	X	Y	Z
Position	-49.56837	-1.65903	422.98041

	Rx	Ry	Rz
Rotation	45.23772	0.00000	90.00000

Now your robot is calibrated, and you can start using it to program and scan new parts.

Note that after you activated the calibration, you can deactivate it, and this will remove all the computed offsets. However, it will not revert the robot to its original position.



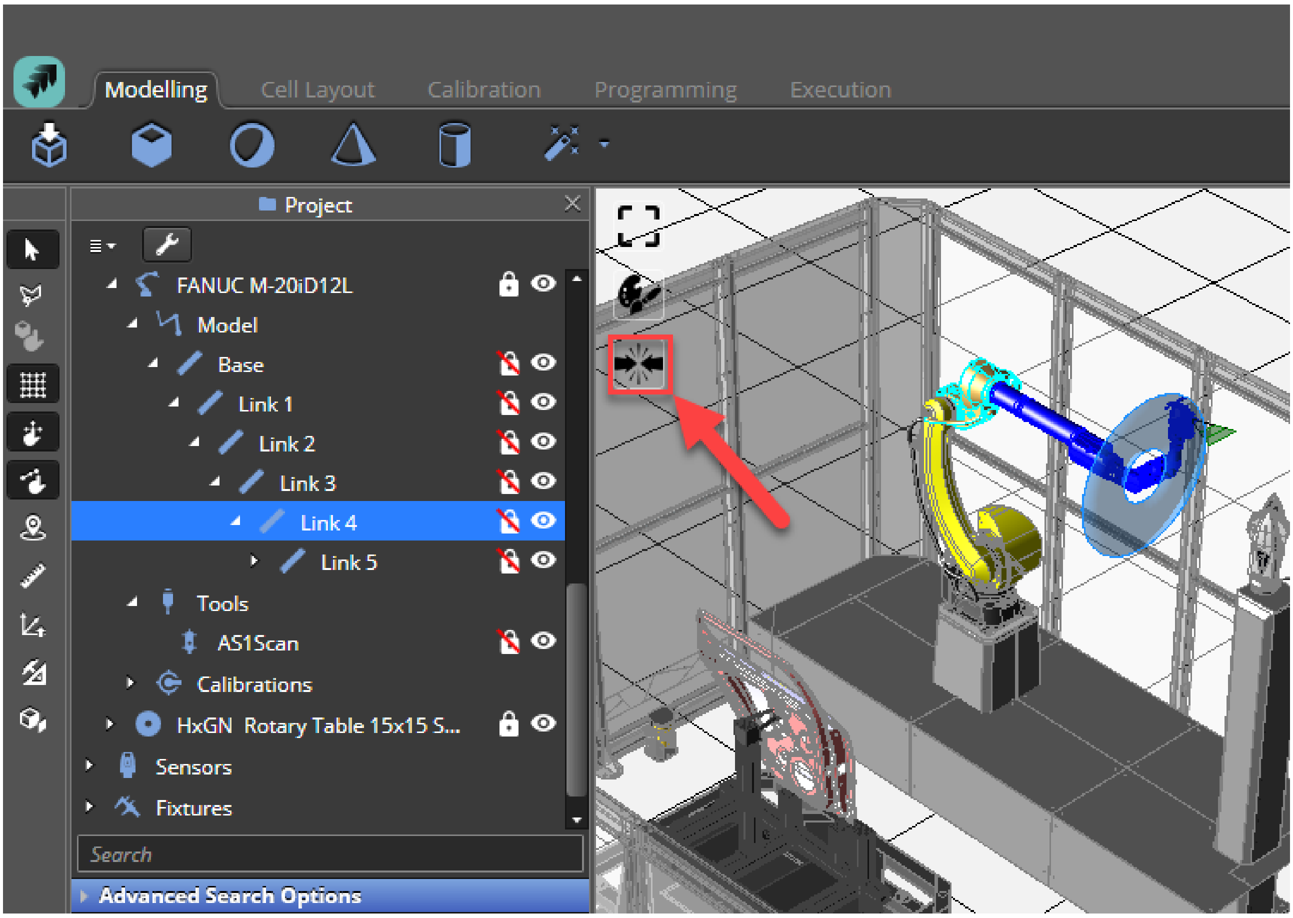
8 Programming

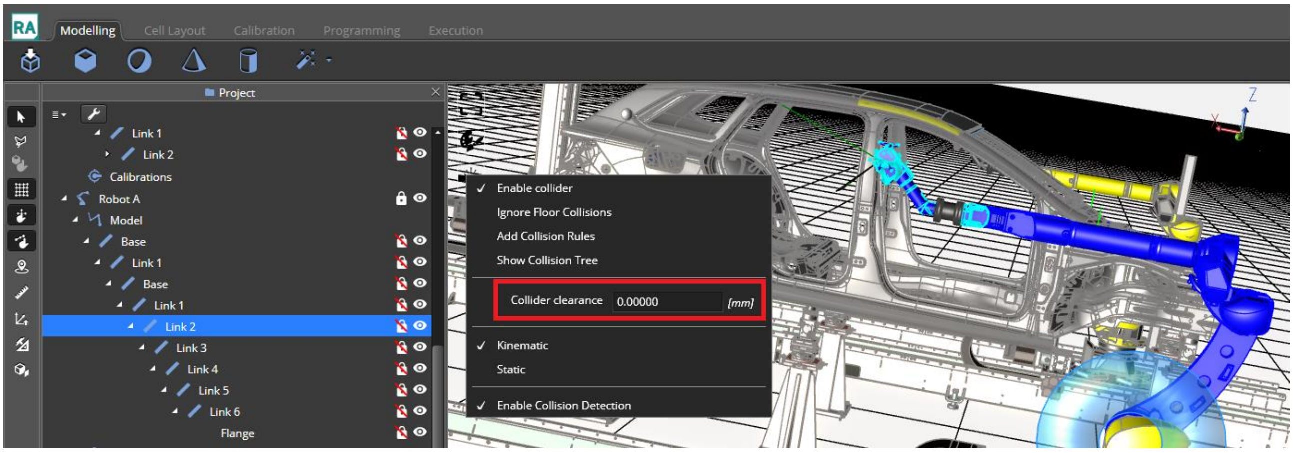
When the cell is ready for work, i.e., the **fixture** and **part** are loaded, properly aligned and the **features** to be measured are available in the project tree, it is possible to program the robot movements needed to inspect the part.

Warning: Make sure to create or import all other objects in your cell. Avoiding collisions is a major part of the planning and so it is crucial for the cell to be precisely modelled.

8.1 Clearance settings

To generate and achieve a safe path, user may use the clearance option to virtually expand the object volume by increasing clearances. User can set a clearance value per each object by clicking on the object and set its value in [mm] by selecting the machine link in the project tree or in the 3D viewer and clicking on the **Collisions** button. The default setting for all elements is 0 [mm].





✓ Enable collider
Ignore Floor Collisions
Add Collision Rules
Show Collision Tree
Collider clearance 0.00000 [mm]
✓ Kinematic
Static
✓ Enable Collision Detection

Info: Adding clearance can be beneficial for collision but should be done carefully since it will limit the motion and area for the planning path. It is recommended to add clearance only for the Tool with value of 20-25 [mm].

It is not recommended to add clearance neither on the part, nor the fixture nor the turn table nor the robot joints. The main risk with adding clearance to Robot joints is internal collision, meaning collision with its own joints. In some cases, user will need to test if there is a collision by moving the joints e.g., if user add clearance of 20 mm on joint 4 and in some positions, there is a collision between joint 4 and joint 6 as shown in **Figure 8-1**.

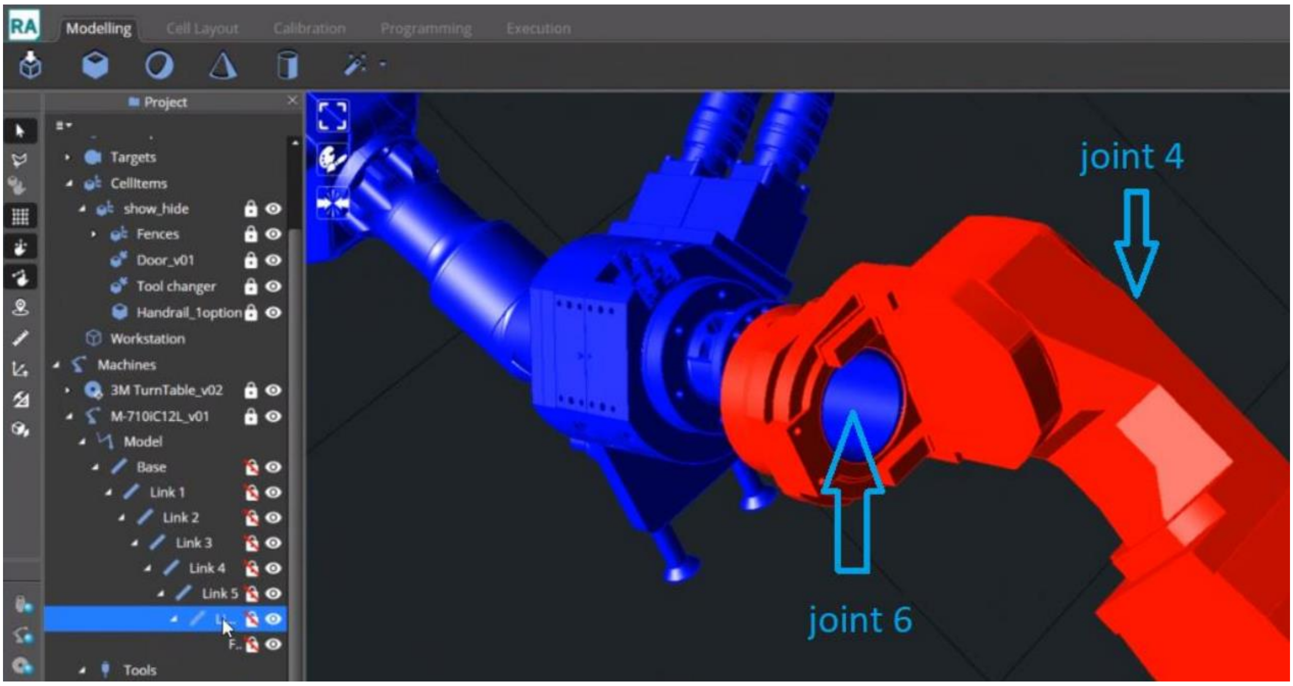


***Figure 8-1**: Collision between Joint 4 and Joint 6 due to an excessive clearance.*

Info: during the automatic path planning, HxGN Robotic Automation adds 10 [mm] clearance to the Tool in addition to the user clearance definition to achieve a safer program and solve discretization issues.

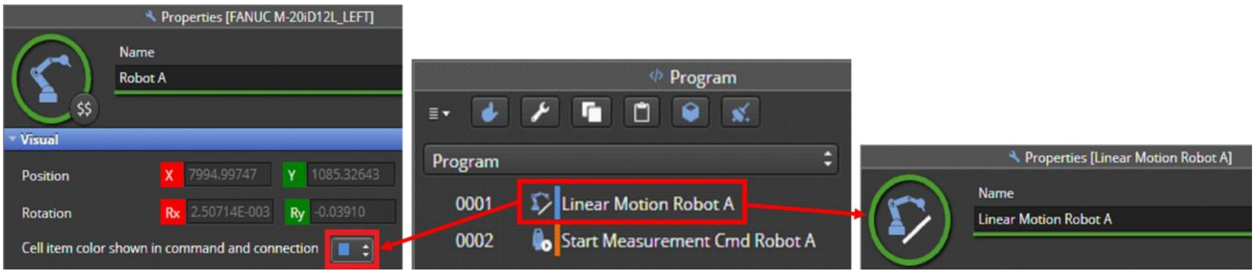
The robot joints clearances are also added 10 mm but only for joints of which the clearance value was changed to a value greater than zero (e.g., if joint 4 clearance is as default zero, it will stay zero, but if user changes it to 2 mm, then actual clearance will be 12 mm).

Warning: in all cases where user changes the default clearances value to a different value than the recommended values, user should move the robot joints within their range and verify there is no internal collision.

8.2 Language syntax

A program in HxGN Robotic Automation is a list of commands that will be executed by the involved cell items either sequentially or in parallel.
Every command is represented by:

1) **ID**: automatically assigned when the command is added.
2) **Icon**: automatically assigned depending on the command type.
3) **Colour bar**: automatically assigned but can be changed in the item properties (see chapter 5.1.6).
4) **Name**: automatically assigned but can be changed in the command properties.



Properties [FANUC M-20iD12L_LEFT]

Name: Robot A

Visual

Position: X 7994.99747 Y 1085.32643

Rotation: Rx 2.50714E-003 Ry -0.03910

Cell item color shown in command and connection

Program

0001 Linear Motion Robot A

0002 Start Measurement Cmd Robot A

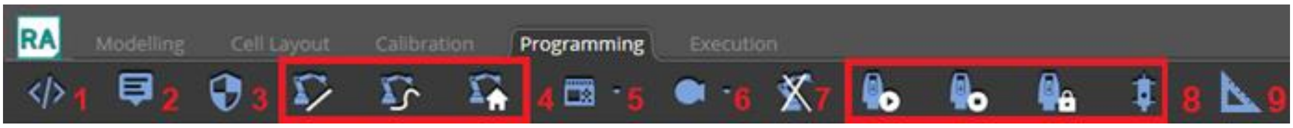
Properties [Linear Motion Robot A]

Name: Linear Motion Robot A

Figure 8-2: Example of two commands and related syntax: the first command has index “0001”, a “Linear Motion Command” icon, the colour line as defined in Robot A properties, and a “Linear Motion Robot A” name defined in the command properties

8.3 Supported commands

The complete list of supported commands is available in the ribbon toolbar within Programming tab.
If there are multiple commands available within a specific command type, a drop-down menu is used to list them.

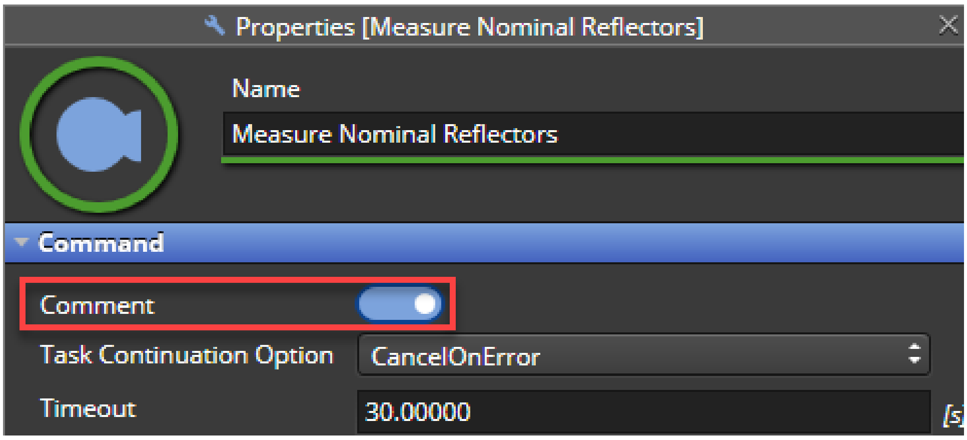


All commands share some common properties:

1) **Name:** will be visualized in the program tree

2) **Command** area having:

a) **Comment:** if the toggle switch is enabled, the command will be skipped in the execution. It is also possible to comment a command by double-clicking it in the program tab.



Properties [Measure Nominal Reflectors]

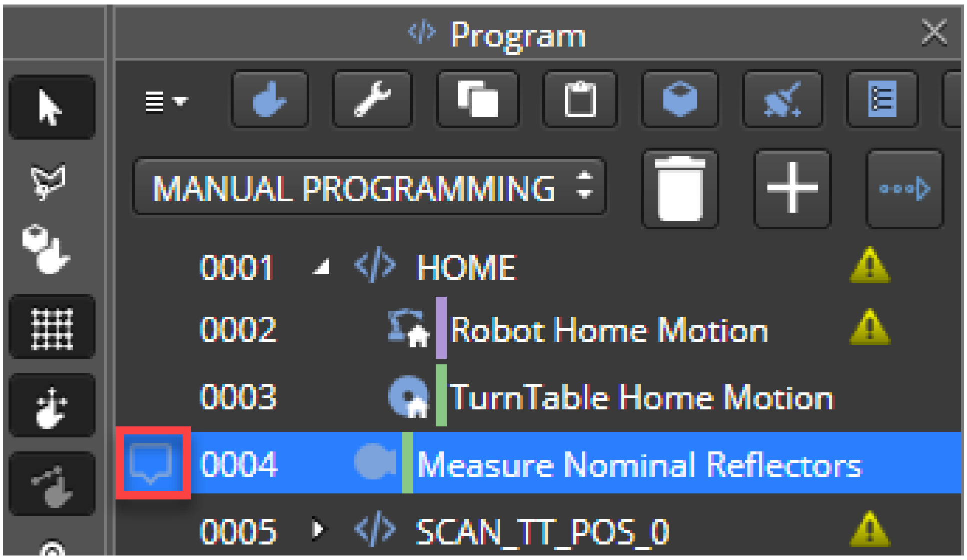
Name: Measure Nominal Reflectors

Command

Comment: ●

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]



0001 HOME

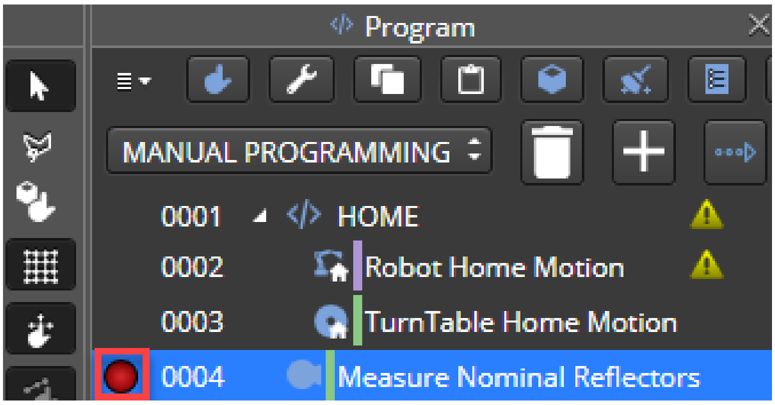
0002 Robot Home Motion

0003 TurnTable Home Motion

0004 Measure Nominal Reflectors

0005 SCAN_TT_POS_0

Info: to set a debug point instead of a comment it is needed to single-click the command line where the execution must stop.



0001 HOME

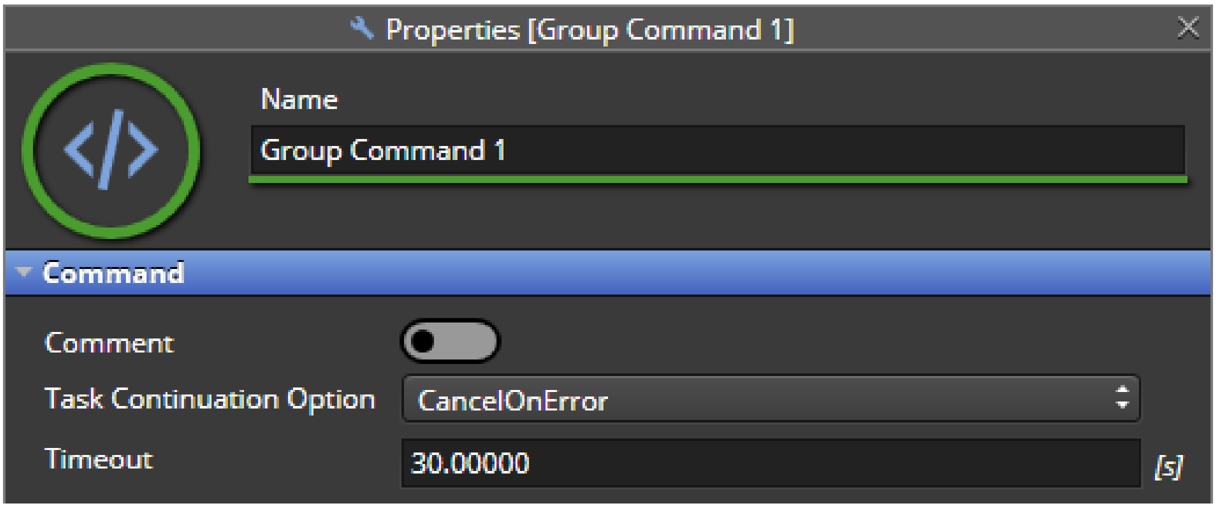
0002 Robot Home Motion

0003 TurnTable Home Motion

0004 Measure Nominal Reflectors

b) **Task Continuation Option:** (not available for Motion commands)
   i) **CancelOnError:** if the commands fails, the program will be immediately terminated with an error
   ii) **ContinueOnError:** if the commands fails, the program will continue
   iii) **Undefined:** if the commands fails, the program will continue

c) **Timeout**: expressed in seconds. If the time defined elapses before the command execution is finished, the execution is terminated with a timeout error.



Properties [Group Command 1]

Name: Group Command 1

Command

Comment

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]

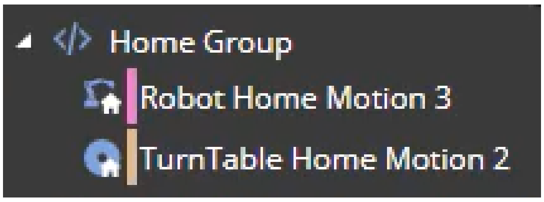
Depending on the command type, there will be dedicated properties as described in the following sections.

8.3.1 Grouping

Grouping (</>) can be used to improve the readability of a program by putting together commands according to a common criterion, e.g., related to a feature scan or to homing movements of all the machines used in the cell.

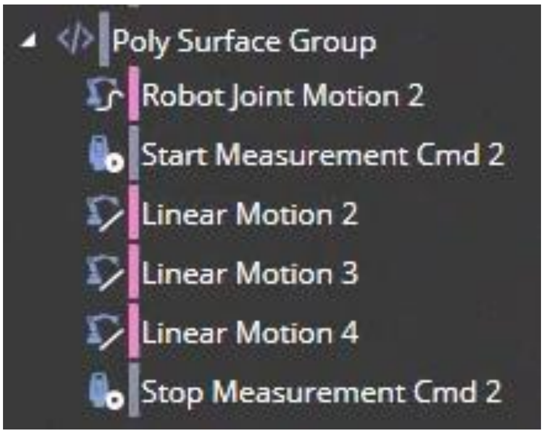
Once grouped, the commands will be executed in sequence, unless the **Parallel Execution** switch is turned on. See chapter Error! Reference source not found. for more information on parallel execution scenarios.

Example – Grouping homing commands.



Home Group
Robot Home Motion 3
TurnTable Home Motion 2

Example – Grouping feature related commands.

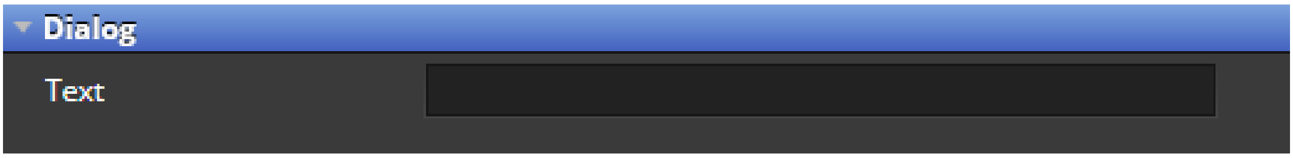


Poly Surface Group
- Robot Joint Motion 2
- Start Measurement Cmd 2
- Linear Motion 2
- Linear Motion 3
- Linear Motion 4
- Stop Measurement Cmd 2

8.3.2 Dialog

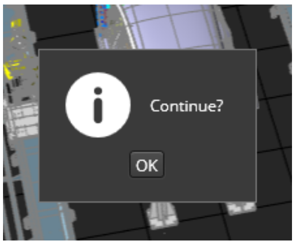
A dialog command (🗨) can be used to display a pop-up window with the text specified in the **Text** property.

User will need to click on **OK** to continue the program execution.



Dialog

Text	

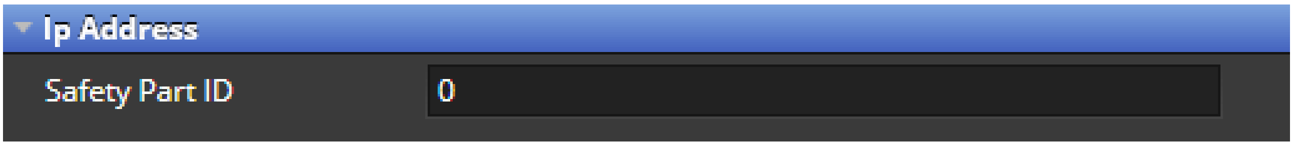


Continue?

Info: dialog commands will not be used in programs executed with the Execution Manager.

8.3.3 Safety

A safety command (🛡) can be used to trigger a specific program represented by its ID (previously configured on the safety device) in the command properties.



Ip Address

Safety Part ID	0

Info: Currently HxGN Robotic Automation only supports Keyence Pokayoke camera system.

8.3.4 Motion

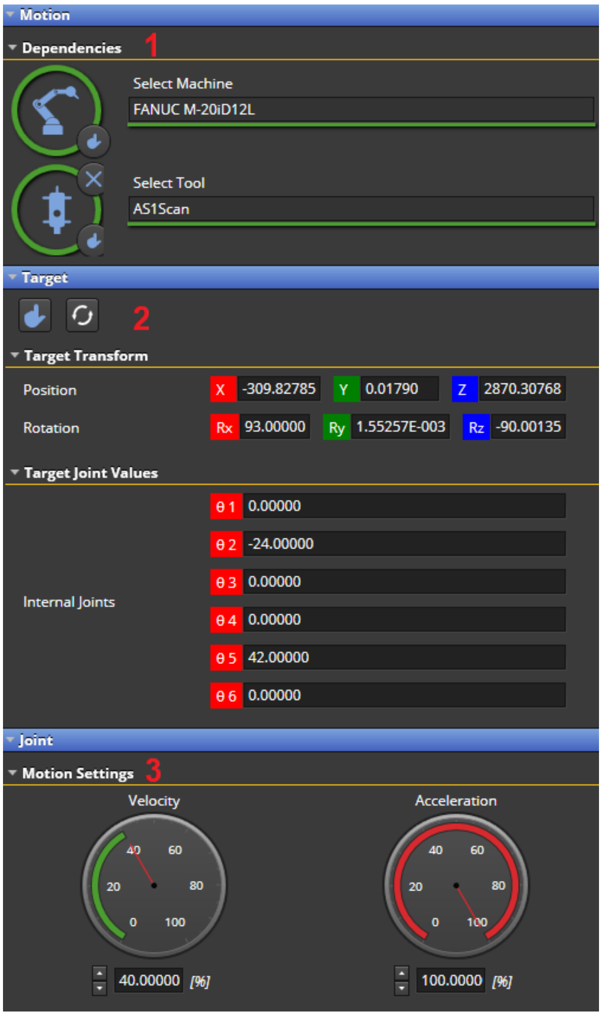
Depending on the selected machine (robots, rotary tables, sliders), the related commands will be shown in the programming bar in this area as explained in the next sections.

All motion commands share the same properties' structure that differs one from another depending on the machine type:

1) **Motion** part has the machines involved in the motion command (e.g., the robot and its tool) that can be selected using the hand icon (☝)

2) **Target** part has the target position to be reached either in cartesian (**Target Transform**) or joint space coordinates (**Target Joint Values**)

3) **Joint/Linear** part has the velocity and acceleration settings for both a joint or a linear movement.



Motion

Dependencies 1

Select Machine: FANUC M-20iD12L

Select Tool: AS1Scan

Target 2

Target Transform

Position: X -309.82785 Y 0.01790 Z 2870.30768

Rotation: Rx 93.00000 Ry 1.55257E-003 Rz -90.00135

Target Joint Values

Internal Joints: θ 1 0.00000; θ 2 -24.00000; θ 3 0.00000; θ 4 0.00000; θ 5 42.00000; θ 6 0.00000

Joint

Motion Settings 3

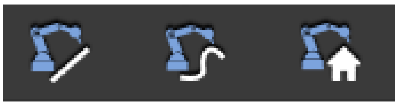
Velocity: 40.00000 [%]

Acceleration: 100.0000 [%]

8.3.4.1 Robot

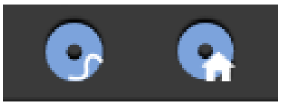
When a robot is selected it is possible to add three types of commands according to the picture below:

1) **Linear Motion**, to specify a cartesian position of the robot.
2) **Joint Motion**, to specify the position of the robot in the joint space- either 7 DOF or 6 DOF depending if the robot is mounted on a slider (see chapter 5.1.3.2) or not (see chapter 5.1.3)
3) **Home Motion**, to move the robot to the defined home position.



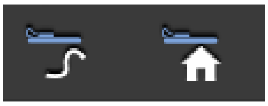
8.3.4.2 Rotary table

The same applies when a rotary table (see chapter 5.1.4) is selected but linear motions are not available, and the joint space is 1 DOF.



8.3.4.3 Slider

The same also applies for sliders (linear joints) where usually a tracker is mounted. In this case depending on if the slider is only horizontal or vertical the joint space will have only 1 DOF, if the slider can move in both vertical and horizontal direction, the joint space will have 2 DOF (see chapters 5.1.1 and 5.1.2).

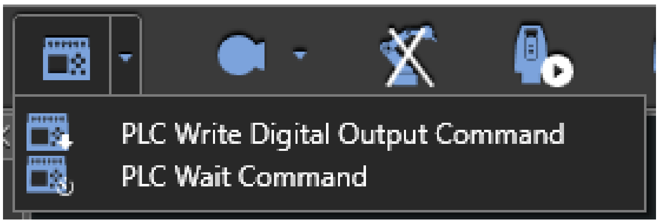


8.3.5 PLC (Programmable Logic Controller)

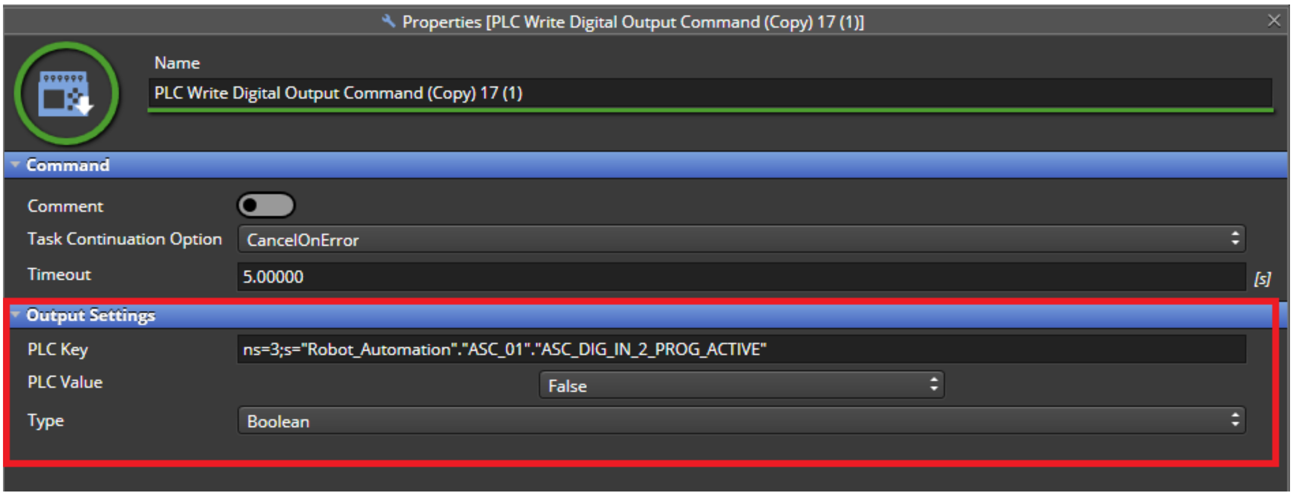
PLC commands are used to communicate with the cell PLC that is responsible to control the overall main logic of the robotic cell and it is the interface with all the other devices connected to it, e.g., motion and safety devices. The communication with the PLC happens through and OPC UA interface where the PLC acts as an OPC UA server and HxGN Robotic Automation as an OPC UA client.

The available commands are:

1) **PLC Write Digital Output**: to be used to set a Boolean value (true or false) to a variable defined in the PLC.
2) **PLC Wait**: to be used to wait that the PLC has received the output signal and the consequent action was successful (e.g. an action is performed on a linked HW device). The program will wait until the PLC communicates the successful handling of the signal.



In both cases the variable settings are done within the **Output Settings** section within the command properties where it is needed to specify the PLC tag to communicate with over OPC UA in the **PLC Key** field.


Properties [PLC Write Digital Output Command (Copy) 17 (1)]

Name: PLC Write Digital Output Command (Copy) 17 (1)

Command

Comment: ○

Task Continuation Option: CancelOnError

Timeout: 5.00000 [s]

Output Settings

PLC Key: ns=3;s="Robot_Automation"."ASC_01"."ASC_DIG_IN_2_PROG_ACTIVE"

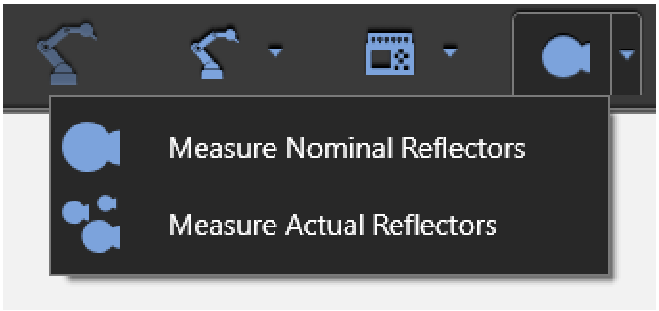
PLC Value: False

Type: Boolean

8.3.6 Compensation

The compensation commands are usually needed in the following cell configurations which are examples of the relationship change between tracker and part:

1) Cell including one or more turn tables, with reflectors mounted on the turn table.
2) Cell including one or more trackers mounted on a slider, with reflectors mounted on some static reference.
3) Cell including both cases above simultaneously.



In general, the compensation flow is composed by two steps: a nominal process measurement followed by an actual process measurement for which dedicated commands are available.

See chapter 8.7 for further information on compensation.

8.3.7 Acquire/release Lock for Multi Robot Synchronization

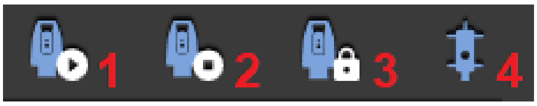
In multi-robot configurations, there may be situations when both robots running in parallel are only allowed to enter a zone one at the time to avoid collisions. To deal with such cases HxGN Robotic Automation makes available for the user the **Acquire/release Lock for Multi Robot Synchronization** command (X).

The essence of this command is that the children of only one of such commands (by means of resource name) must be executed at the same time so whichever command (e.g., Lock Resource A) locks a resource which is specified in the *Multi Robot Synchronization* | *Resource Name* field will execute; the other commands sharing the resource (e.g., Lock Resource B, Lock Resource C, and so on) will wait for the Lock Resource A execution to be over.

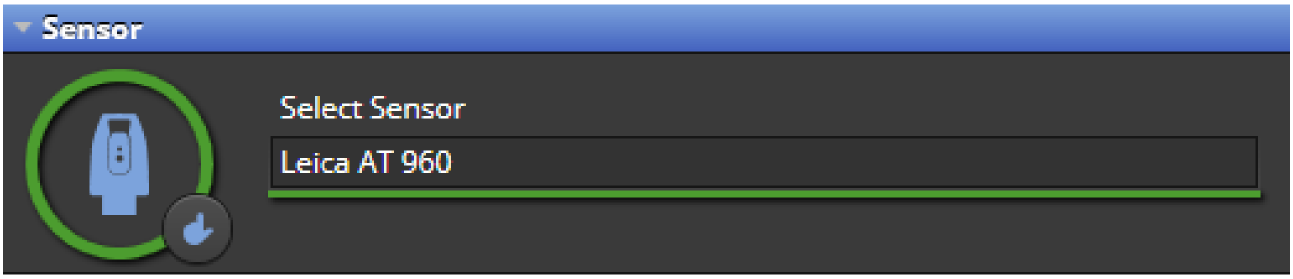
Pro Tip: programmers can see the analogy between this command and a C#'s "lock" or a C++ "critical section" keywords, which allow only one thread to enter the part that's locked, and the lock is not shared with any other processes.

8.3.8 Sensors

In this area the user can select the commands to control the Trackers (e.g., AT960) and Laser scanners in use (e.g., AS1) in the project.



All tracker commands (start/stop measurement, lock on) must specify the sensor to be used in the **Sensor** section of the properties (e.g., AT960).



Sensor

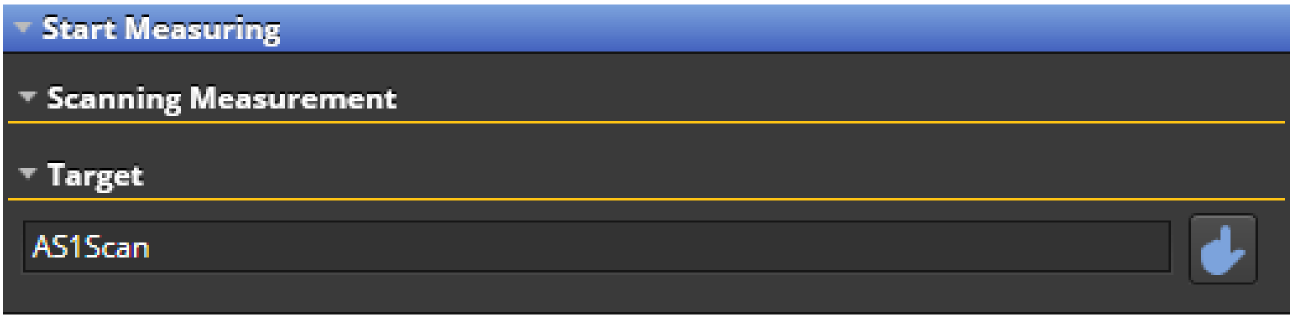
Select Sensor

Leica AT 960

Depending on the command, they have specific parameters to define the behaviour as explained below.

8.3.8.1 Start Measurement

In the Start Measurement command the tool to scan must be specified under *Start Measuring* | *Target* section.



Start Measuring

Scanning Measurement

Target

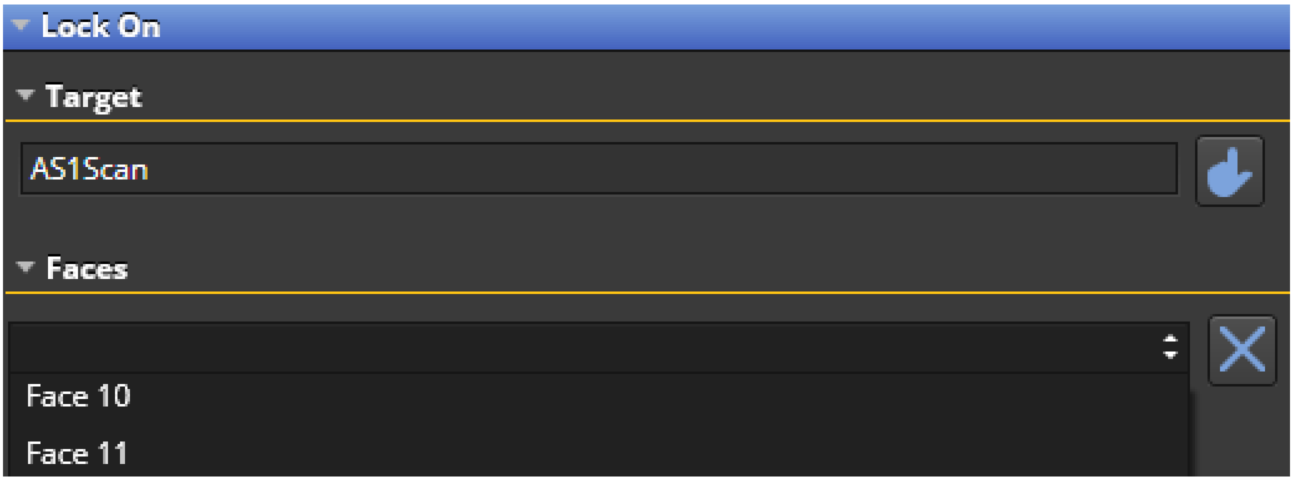
AS1Scan

8.3.8.2 Stop Measurement

In the Stop Measurement there is no need to specify any properties as the stop command will refer to the earlier Start Measurement command settings.

8.3.8.3 Lock On

A Lock On command (🔒) can be used to force the tracker to lock on a specific target that can be either the face (to be defined under *Lock On* | *Faces* using a drop-down menu) of a scanner (to be defined under *Lock On* | *Target*) or a reflector.



Lock On

Target: AS1Scan

Faces

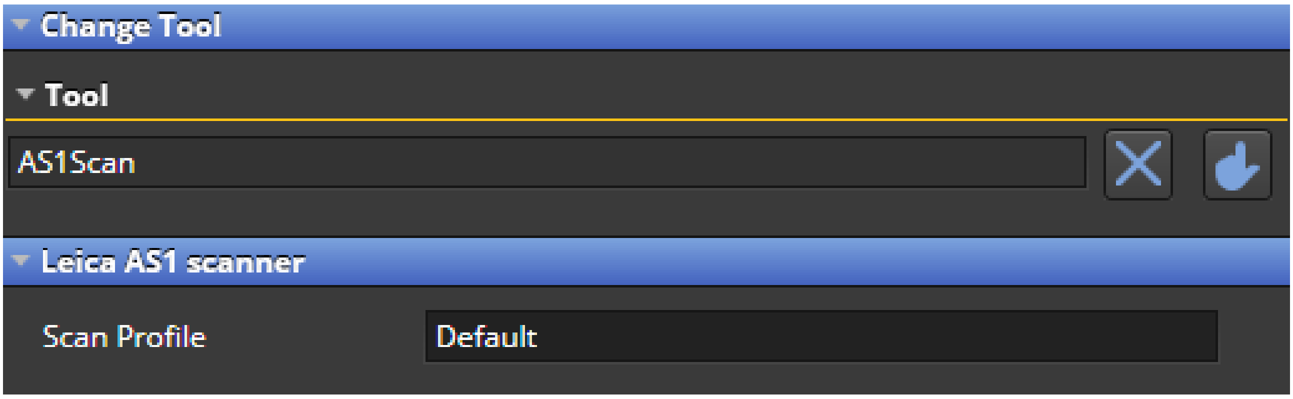
Face 10

Face 11

8.3.8.4 Change AS1 Settings

A Change AS1 Settings command (🔧) can be used to change the scanning profile of a laser scanner by specifying the Scan Profile in the corresponding field.

Warning: Scan profile must be set in RDS and the name must match exactly the one specified in the command **Scan Profile** property



Change Tool

Tool: AS1Scan

Leica AS1 scanner

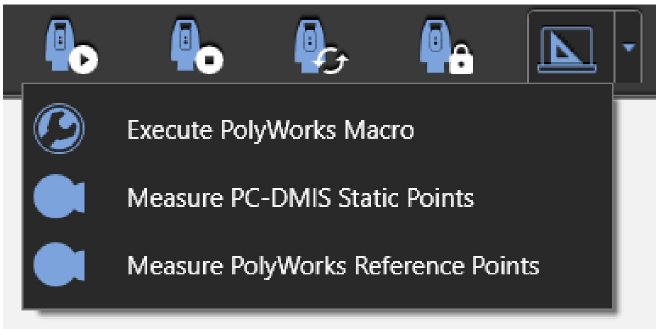
Scan Profile: Default

8.3.9 Metrology

Metrology commands group contains a list of commands which are specifically developed for special functionalities available in the interface with PC-DMIS and PolyWorks. **Measure PC-DMIS Static Points** is specific to PC-DMIS and it is explained in chapter 10.1.2.

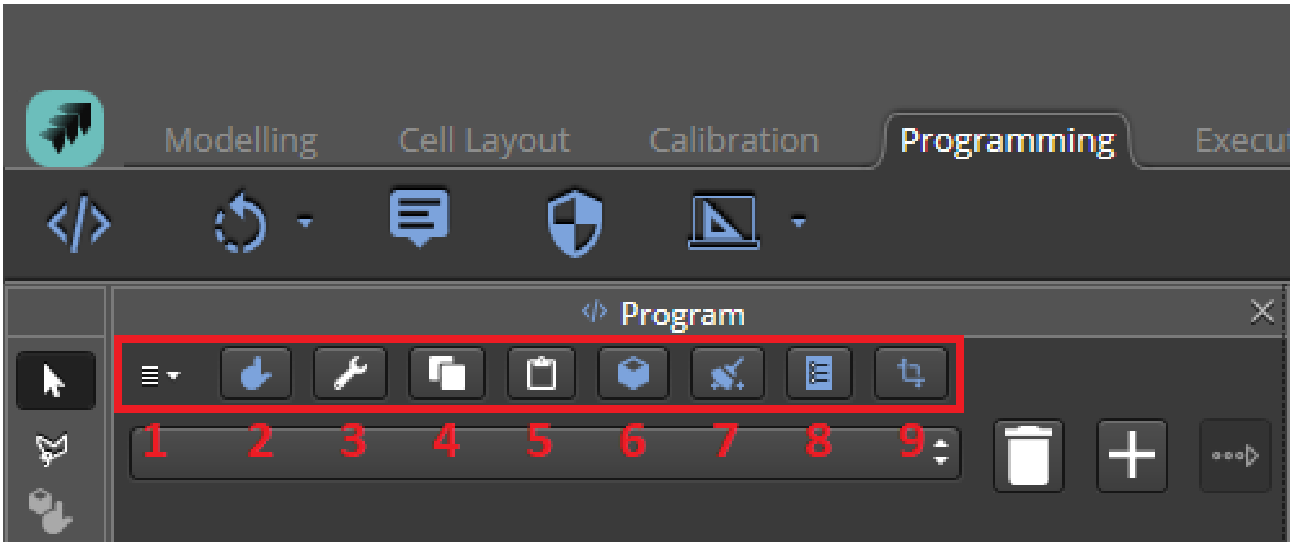
Execute PolyWorks Macro and **Measure PolyWorks Reference Points** are specific to PolyWorks. Please refer to chapter 10.3.1.1 for the explanation on this command usage and properties.

Depending on the Metrology software configured in the project, the commands will be visible or hidden.


Execute PolyWorks Macro
Measure PC-DMIS Static Points
Measure PolyWorks Reference Points

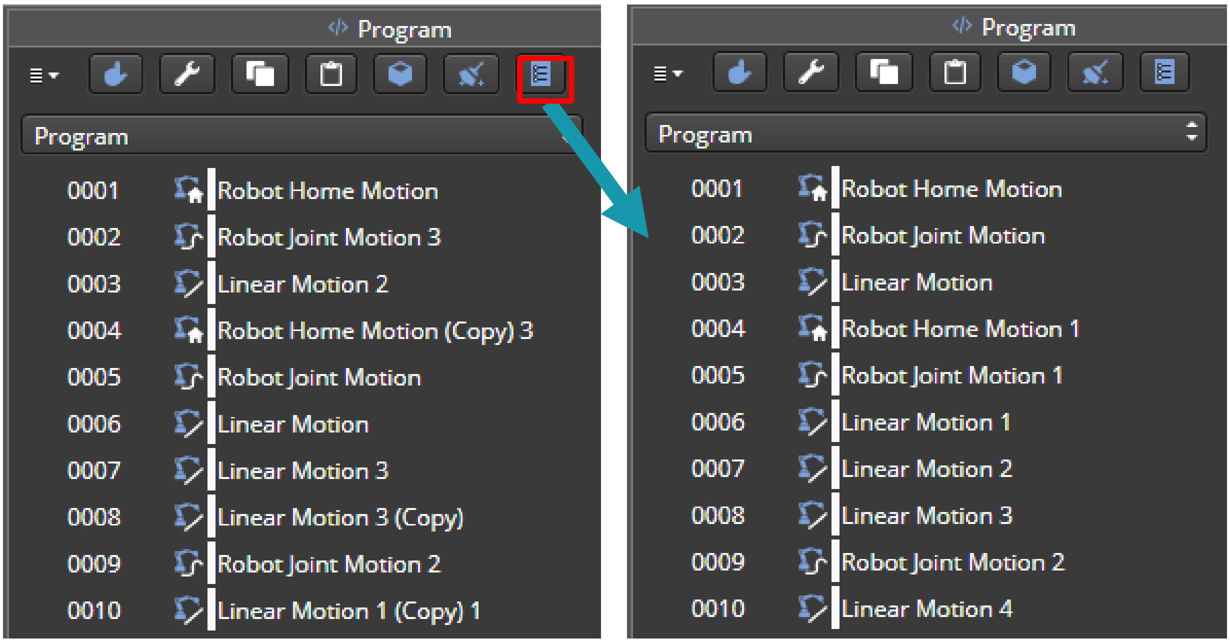
8.4 Programming tools [from V2.0.2]

A toolbar above the program can help the user with complex programs.



1) **Collapse/Expand all** the commands in the program.
2) **Select All** the program and all its commands and subroutines.
3) **Program Properties**   the program properties.
4) **Copy** the selected commands (and subroutines) of the program (to afterwards paste them in another program).
5) **Paste** previously copied commands and subroutines into a new program (will be pasted after the current program if there are already commands and subroutines in place).

6) **Paste After** the previously copied commands and subroutines into a new program (will be pasted after the selected command or subroutine). This choice will allow you to paste a program to a specific subroutine folder for example.
7) **Clear Highlighting** in case the program stopped in error and the last active command will stay highlighted after debugging.
8) **Beautify Command Numbering** in case the name and numbering are confusing, it is possible to refactor the commands that will be renamed and renumbered correctly (see example below).



	Program
0001	Robot Home Motion
0002	Robot Joint Motion 3
0003	Linear Motion 2
0004	Robot Home Motion (Copy) 3
0005	Robot Joint Motion
0006	Linear Motion
0007	Linear Motion 3
0008	Linear Motion 3 (Copy)
0009	Robot Joint Motion 2
0010	Linear Motion 1 (Copy) 1

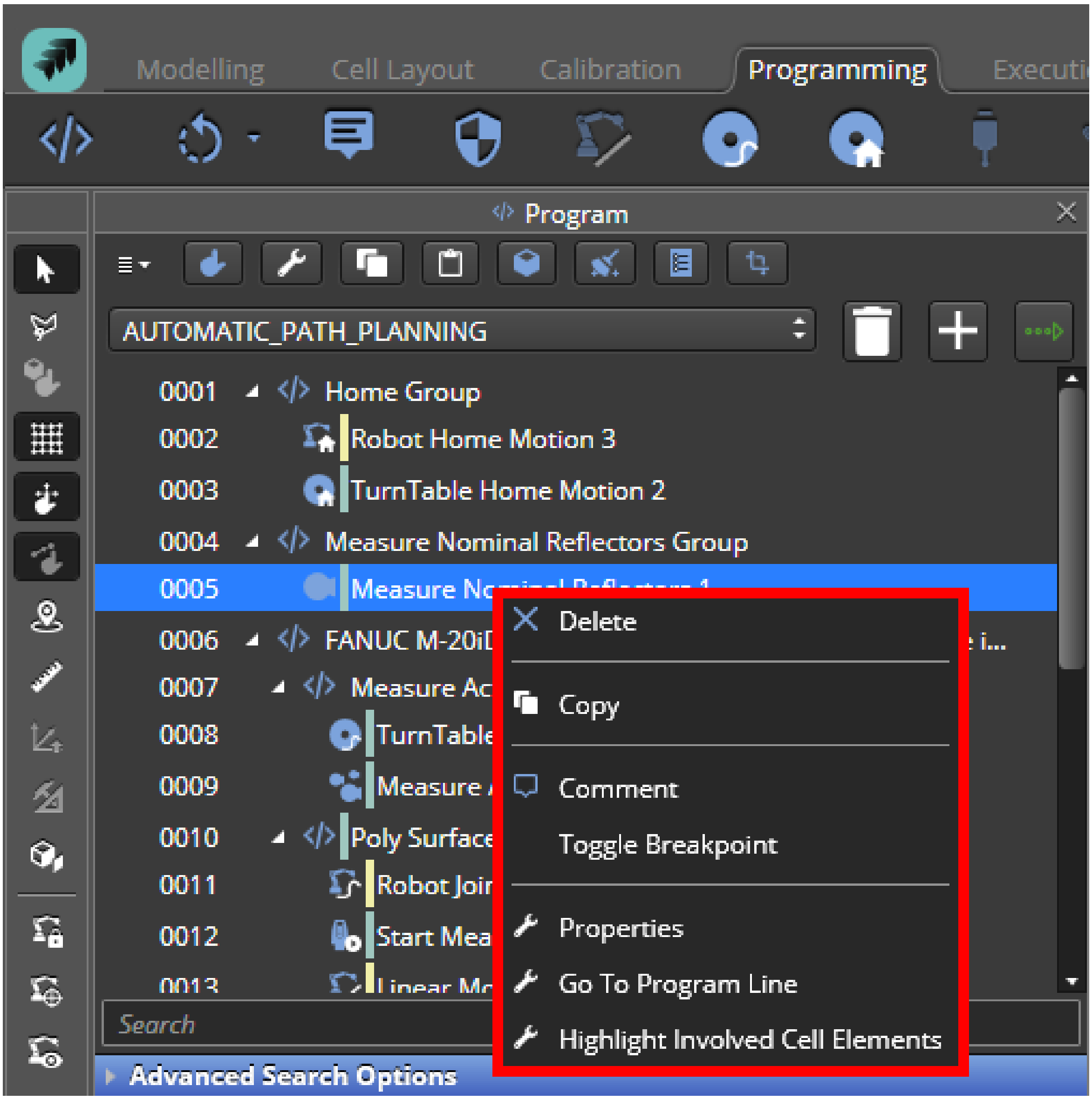
	Program
0001	Robot Home Motion
0002	Robot Joint Motion
0003	Linear Motion
0004	Robot Home Motion 1
0005	Robot Joint Motion 1
0006	Linear Motion 1
0007	Linear Motion 2
0008	Linear Motion 3
0009	Robot Joint Motion 2
0010	Linear Motion 4

9) **Automatic Program Adjustment** re-calculates an existing program when there is a change in the model. This allows the user to migrate old programs according to a new model. Please refer to chapter 0 for a complete explanation of the feature.

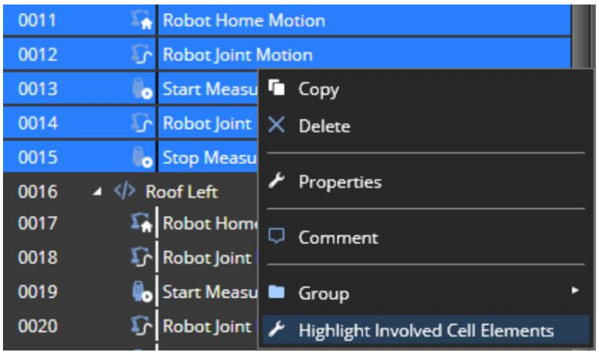
8.5 Generic command menu

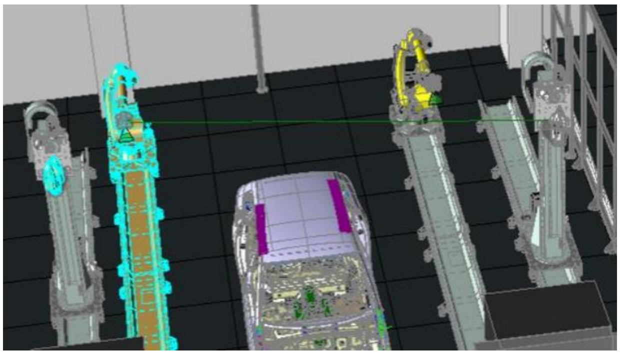
By right clicking on a command it is possible to see the command menu that will show for all commands the following options:

1) Delete the command
2) Copy the command
3) Comment: disable command execution, see chapter 8.3
4) Toggle Breakpoint: set a debug breakpoint, see chapter 8.3
5) Properties: open properties panel on the right
6) Go To Program Line: jump to the selected program line



7) Highlight Involved Cell Elements: alternatively this can be done by double click on a program command line. This will highlight the cell items involved in that part of the program. The user can select single commands, multiple commands or even subroutines. The highlight is cancelled by pressing the [Esc] button, running validation or executing the program.

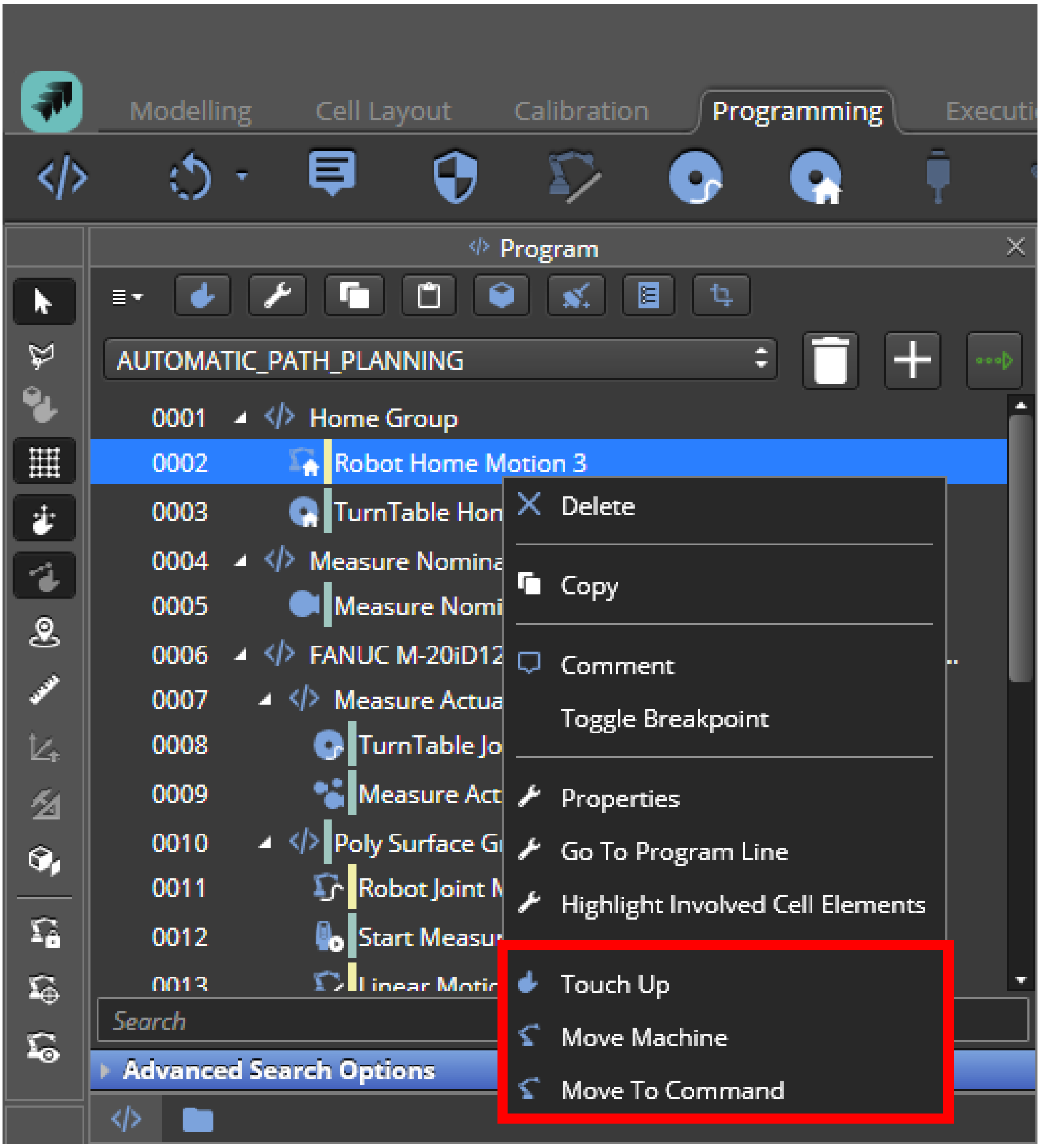




8.6 Motion command menu

Motion commands have additional options to be selected:

1) **Touch Up**: update the command with the actual position of the machine that was moved in the 3D view with respect to the previously defined position. This can be used to fine-tune the positions without the need to create new commands or change the program workflow.
2) **Move Machine**: send the machine involved in the movement (e.g. robot) to the position where the command will be executed
3) **Move To Command**: same as above but considering all the machines involved in the movement (e.g. both robot and rotary table). Using this mode will avoid causing possible collisions in the cell among the involved machines.



8.7 Compensation

The idea of the compensation approach is to make it as agnostic as possible to the cell setup. This is the generic compensation used in RA to compensate for relative motions of cell items causing the COP collected in different scans to be misaligned with each other whenever the relationship between tracker and part is changed.

This situation can happen in two cases:
1) Dynamic reflector scenario: the part moves, so the reflectors change position accordingly (e.g. reflectors linked to the part). See chapter 8.13.1 for a notable example.
2) Static reflector scenario: the tracker moves from its previous position (e.g. tracker moving on a slider). See chapter 8.13.2 for a notable example.

Compensation has been built with the following flow:

- The reflectors are measured at home position (**nominal reflectors**): this will allow the tracker to create an internal compensation map and adjust for any changes in the relationship between the tracker and the part

- Weather there is a change in the relationship between the tracker and the part, e.g. either the turn table or the slider move, this requires the tracker to understand what was changed compared to its earlier compensation table with respect to the current turn table or tracker position.

- The reflectors are measured in the new position (**actual reflectors**): this will allow the tracker to update the compensation table that it was created with the nominal reflector positions before and the newly measured actual reflector positions and apply a best fit alignment.

It is always recommended to scan all the reflectors in the nominal position. It is not possible to measure all the reflectors from a single position, it is needed to move the turn table (or the slider) and perform another nominal measurement to collect the measurement of all the reflectors and include them into the compensation map together with the reflectors already measured.

After each move, either measuring nominal or actual reflectors, at least three of the original nominal reflector positions need to be included to reduce the best-fit alignment error at minimum. The more moves are used to measure the reflector positions, the more noise will be introduced in the best-fit alignment process.

Info: Path planning algorithm already embeds the above logic and create programs that include this compensation flow.

8.7.1 Measure Nominal Reflectors workflow

1) The home position is first considered to measure the reflectors as the first pose.

2) The user adds a **Measure Nominal Reflectors** command and HxGN Robotic Automation tries to measure all the nominal reflector positions. If in the home position it is not possible to measure enough reflectors, a warning is shown during the program execution in the status window. This would imply that the user needs to create a new position where the reflector is visible or move the reflector to be able to be measured from the home position.

3) If all reflectors can be measured in the home position, then the overall program will be made of a the following instructions:
   a) Move the Turn Table or Slider to Home
   b) Measure Nominal Reflectors

4) If in the home position only some of the reflectors can be measured, a set of turn tables or slider positions need to be added in the program to measure all reflectors so the program will have a sequence of instructions to execute more turn table or slider positions and measure nominal reflector positions in the new turn table or slider position

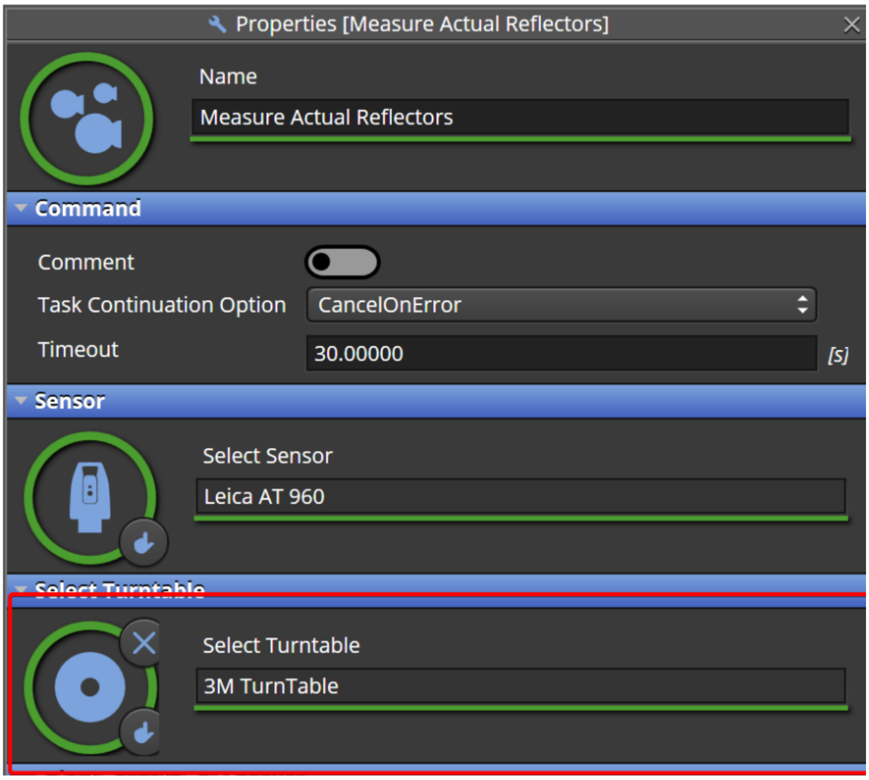
5) The amount of additional turn table or slider position depends on the specific situation, e.g., if three pose are defined either for a Turn Table or a Slider, the sequence of instruction will have the following instructions:
   a) Move the Turn Table or Slider to Home
   b) Measure Nominal Reflectors
   c) Move the Turn Table or Slider to Pose 1
   d) Measure Nominal Reflectors
   e) Move the Turn Table or Slider to Pose 2
   f) Measure Nominal Reflectors
   g) Move the Turn Table or Slider to Pose 3
   h) Measure Nominal Reflectors

Notable use cases of the compensation usage are described in chapter 8.13.

8.7.2 Measure Actual Reflectors workflow

Before starting any measurement routine, if the reflectors have moved compared to the position they had when measured by a **Measure Nominal Reflectors** command, it is needed to measure the actual position of at least three of them. For this purpose, the **Measure Actual Reflectors** command must be used with the same considerations described in the previous chapter.

Among the properties it is possible to select a turn table where the reflectors are mounted on. Otherwise, the field must be kept empty (e.g., when the turn table does not move, and the measure actual command is needed because of a tracker movement instead).

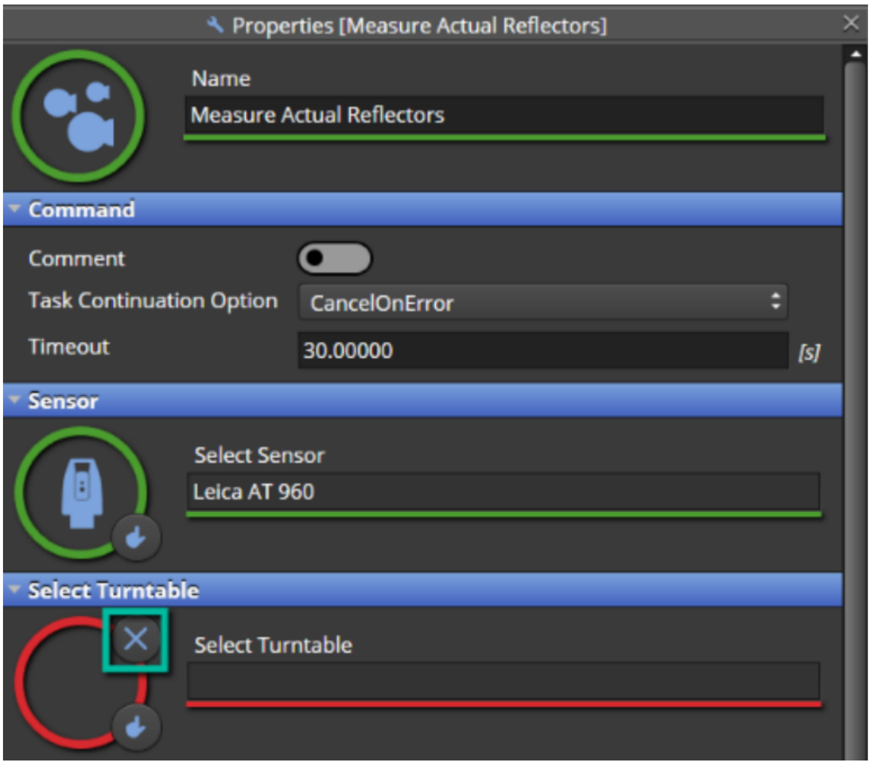

Properties [Measure Actual Reflectors]

Name: Measure Actual Reflectors

Command
- Comment
- Task Continuation Option: CancelOnError
- Timeout: 30.00000 [s]

Sensor
- Select Sensor: Leica AT 960

Select Turntable
- Select Turntable: 3M TurnTable



Properties [Measure Actual Reflectors]

Name: Measure Actual Reflectors

Command

Comment: ○

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]

Sensor

Select Sensor: Leica AT 960

Select Turntable

Select Turntable:

It is always recommended to measure as many reflector positions as possible in every turn table or slider position to have the optimal best-fit alignment process.

8.7.3 Evaluating the Quality of the Compensation

The assessment of the quality of the compensation is done at two stages:

1) If the program has several Measure Nominal Reflectors commands, it assesses the quality of the newly measured reflectors with respect to the reflectors measured in the previous Measure Nominal Reflectors commands.

2) Assesses the quality of the reflectors on each Measure Actual Reflectors command.

At each stage, the outcome of the assessment is either that a particular reflector needs to be disabled from the command (Measure Nominal Reflectors command or Measure Actual Reflectors command) because it may add too much noise during measurement or discard the command and measure at a different pose.

8.7.3.1 Measure Nominal Requirements

- The first Measure Nominal Reflectors command must measure at least 3 reflectors.
- The next Measure Nominal Reflectors command must measure at least 3 reflectors in common with the first Measure Nominal Reflectors command.
- It is suggested to use as few Measure Nominal Reflectors command as possible to measure as many reflectors as possible.

The procedure is to execute the command to assess the quality of compensation data.

Once executed, the first Measure Nominal Reflectors command will not show any information about the quality of the best fit (is not computed on this case) and must be ignored.

The rest of Measure Nominal Reflectors commands will show a chart with information about the result of the best fit.

This chart must be used to assess the quality of the measured data as explained in chapter 8.7.3.2.

8.7.3.2 Understanding alignment data

Considering **Figure 8-3** the following areas are to be evaluated:

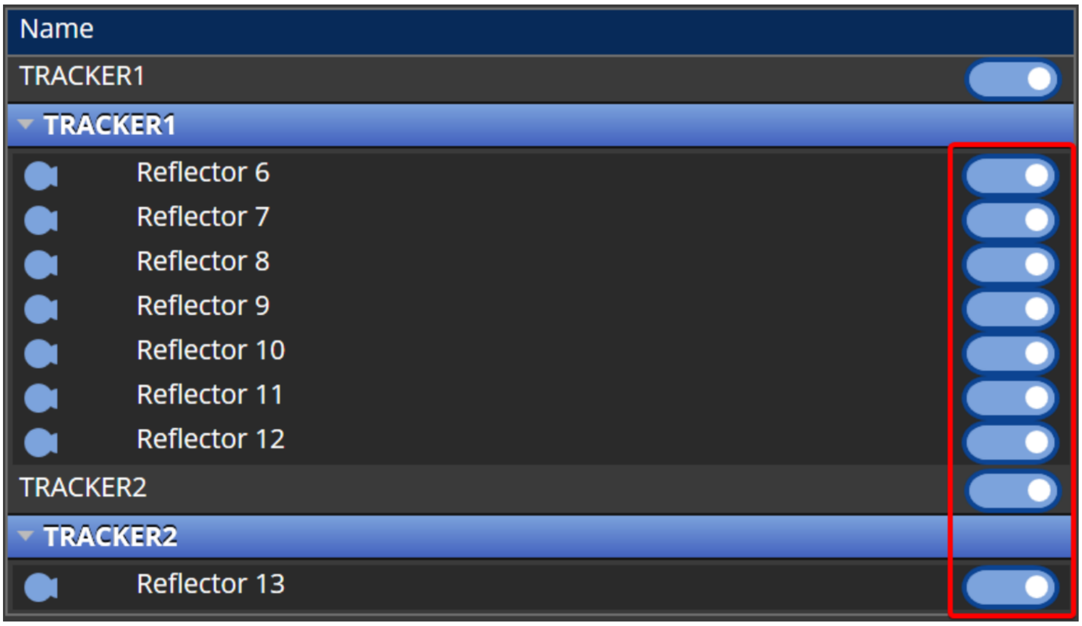
1) **Region 1:** given a particular reflector, this area of the graph shows the absolute error (distance) between its measured position at the first Measure Nominal Reflectors command pose and the estimated position resulting from applying the computed compensation to the reflector measured at the current Measure Nominals Reflectors command pose.

2) **Region 2:** this area shows the new measured reflectors that will be available for future Measure Nominal Reflectors command. No error can be computed on them because they are not used for best fit computation.

3) **Region 3:** this area gives some statistic summary of the best fit computation.
   a) Average between the absolute errors of all the reflectors of region 1.
   b) RMS between values of region 1 values.
   c) Max absolute error (distance) of region 1 values.

If the result doesn't fit with the expected quality, there are two main actions that can help to improve the quality of the compensation:

1) The absolute error of one of the measurements is significantly greater than the Average/RMS: in this case it is possible to disable the measurement of this reflector.



Name

TRACKER1

TRACKER1

Reflector 6

Reflector 7

Reflector 8

Reflector 9

Reflector 10

Reflector 11

Reflector 12

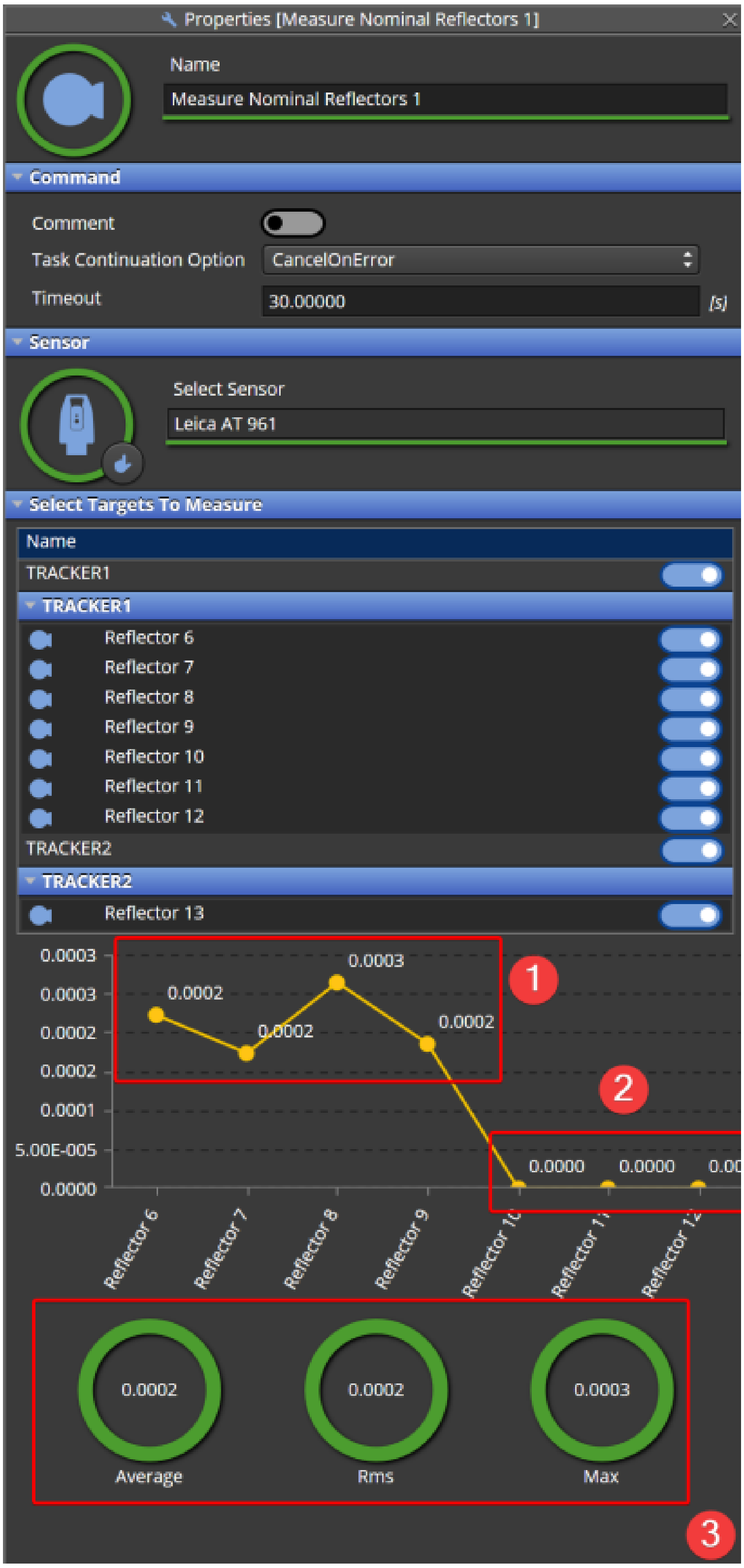
TRACKER2

TRACKER2

Reflector 13

2) High Average/RMS values but there is not a clear measurement with a significantly greater error than the Average/RMS: in this case it is recommended to try another position.

Once the corrective actions are performed it is necessary to execute and assess the quality again.



Properties [Measure Nominal Reflectors 1]

Name: Measure Nominal Reflectors 1

Command

Comment: ○

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]

Sensor

Select Sensor: Leica AT 961

Select Targets To Measure

Name	
TRACKER1	●
TRACKER1	
Reflector 6	●
Reflector 7	●
Reflector 8	●
Reflector 9	●
Reflector 10	●
Reflector 11	●
Reflector 12	●
TRACKER2	●
TRACKER2	
Reflector 13	●

Average	Rms	Max
0.0002	0.0002	0.0003

***Figure 8-3**: Compensation quality chart*

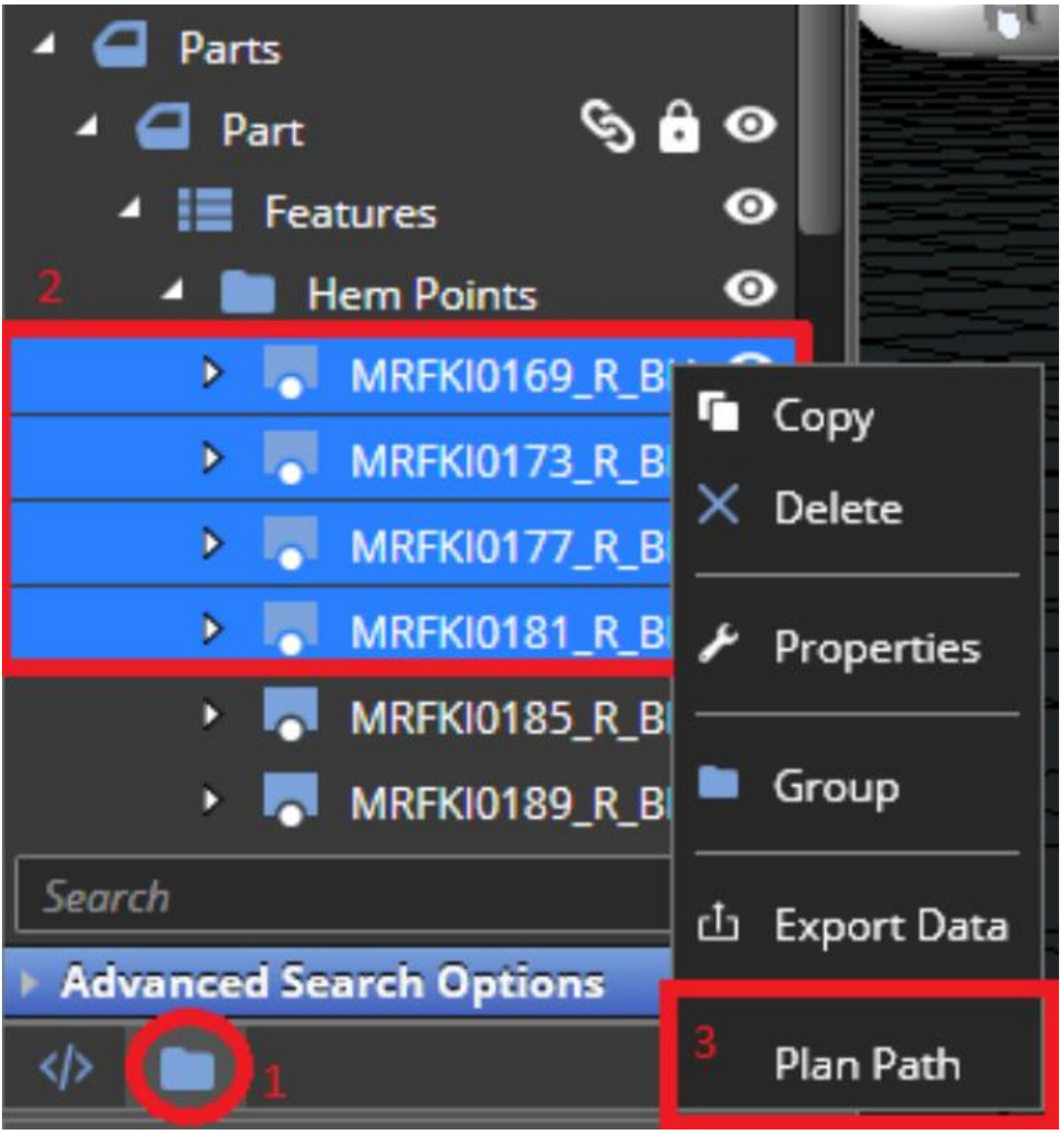
8.7.3.3 Measure Actual Requirements

Each Measure Actual instruction should measure at least three reflectors in common with the measured reflectors with Measure Nominal instructions. It is recommended to try to find positions where there are as much reflectors as possible. On each Measure Actual command will show a chart with information about the result of the best fit. You can use this data to assess the quality of the measured data as explained in chapter 8.7.3.2.

8.8 Automatic mode programming – Feature path planning

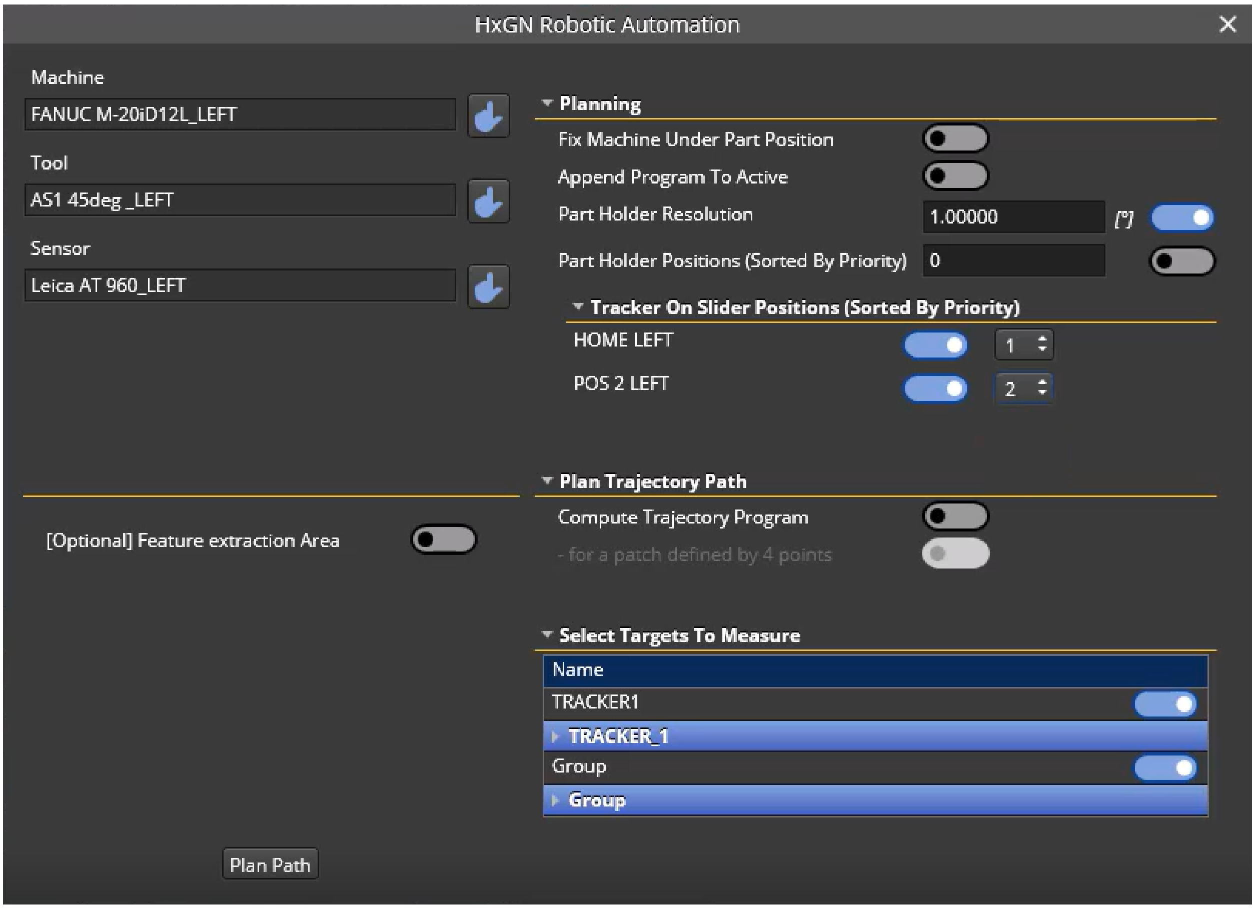
Automatic mode allows the user to automatically create a program based on the metrology features to be measured.
To plan a path, the following actions must be done:

1) Select the Project tab in the upper left window.
2) Select the features you want to measure and right click.
3) Click on the **Plan Path** option.



.

4) A pop-up window will appear where path planning process settings can be defined. The user will need to first select the **Machine**, **Tool** and **Sensor** to be used by clicking on the relative **hand** icons and then select them in the 3D view.



HxGN Robotic Automation

Machine: FANUC M-20iD12L_LEFT

Tool: AS1 45deg _LEFT

Sensor: Leica AT 960_LEFT

[Optional] Feature extraction Area: off

Planning

- Fix Machine Under Part Position: off
- Append Program To Active: off
- Part Holder Resolution: 1.00000 [°] on
- Part Holder Positions (Sorted By Priority): 0 off

Tracker On Slider Positions (Sorted By Priority)

- HOME LEFT: on, 1
- POS 2 LEFT: on, 2

Plan Trajectory Path

- Compute Trajectory Program: off
- - for a patch defined by 4 points: off

Select Targets To Measure

Name	
TRACKER1	on
TRACKER_1	
Group	on
Group	

5) Under **Planning** there are different options:

a) **Fix Machine Under Part Position** toggle switch:

i) by default, is checked out. The planning path for the selected features will try to find best positions for the Turn Table, with minimal positions for covering the features.

ii) If checked in, HxGN Robotic Automation will lock the rotary table holding the part and the path planning for the selected features will be done only for the current rotary table position so the path planning algorithm will not try to find the best rotary table positions for the scan. Before clicking on **Plan Path**, user can change the current Turn Table to the desired position.

b) **Append Program To Active** toggle switch**:**

i) By default, it is checked out. Planning path will generate new program.

ii) If checked in, the generated planning path will not generate a new program, instead it will append it to the Active program if there is one (the user can activate a program by selecting it using the relative drop-down menu within the Program window). In this way the path planning algorithm will try to combine different scans covering the relative features into one single scan.

c) **Part Holder Resolution:** defines the intervals in which the machine under the part can move (e.g. a rotary table). By default, the toggle switch is checked in with 1 degree resolution. User can limit the search resolution to plan a path for what are considered best Turn Table positions.
E.g., if 45° is used as resolution, path planning will allow 0°, 45°, 90°, 135°, 180°, etc positions. If user defines 60° as resolution, only 0°, 60°, 120°, 180°, etc positions will be allowed. Otherwise, the path planning algorithm will consider any position between 0° and 360° as possible positions (e.g. 42°, 173°, and so on).

d) **Part Holder Positions:** by default, the toggle switch is checked out. User can specify specific positions for the Turn Table, e.g. 42°, 87°, 123° degrees. The planning will not calculate the optimal turn table positions but it will use only the positions provided by the user (or part of them) to compute the program.

6) Under **Tracker On Slider Positions (Sorted By Priority)**- optional if a movable tracker on slider is used in the cell- it is possible to select which saved fixed positions (see chapter 8.8.2) will be considered by the path planning algorithm:

a) Enable the desired positions for the slider with the tracker that are to be used for the measurements of the features on the part with the relative toggle switches.

b) Set the priority of each position (the smaller the digit, the higher the priority, e.g. 1 has higher priority than 2). If two positions have the same priority, the top one will be used first. The path planning algorithm will try to generate a program that scans as many features as possible using the highest priority slider position, and continue with other positions to scan the remaining features if any.

c) The enabled positions will not be used for the compensation part of the path planning algorithm. The slider positions needed for the Measure Nominal Reflectors commands are computed automatically by the path planning algorithm to generate an optimal set of slider positions that allows to measure all the reflectors. The enabled positions are used only for the measurement part of the program.

d) If the current slider position is to be used to make the measurements, none of the positions must be enabled. Before clicking on **Plan Path**, the user can change the slider to the desired position.

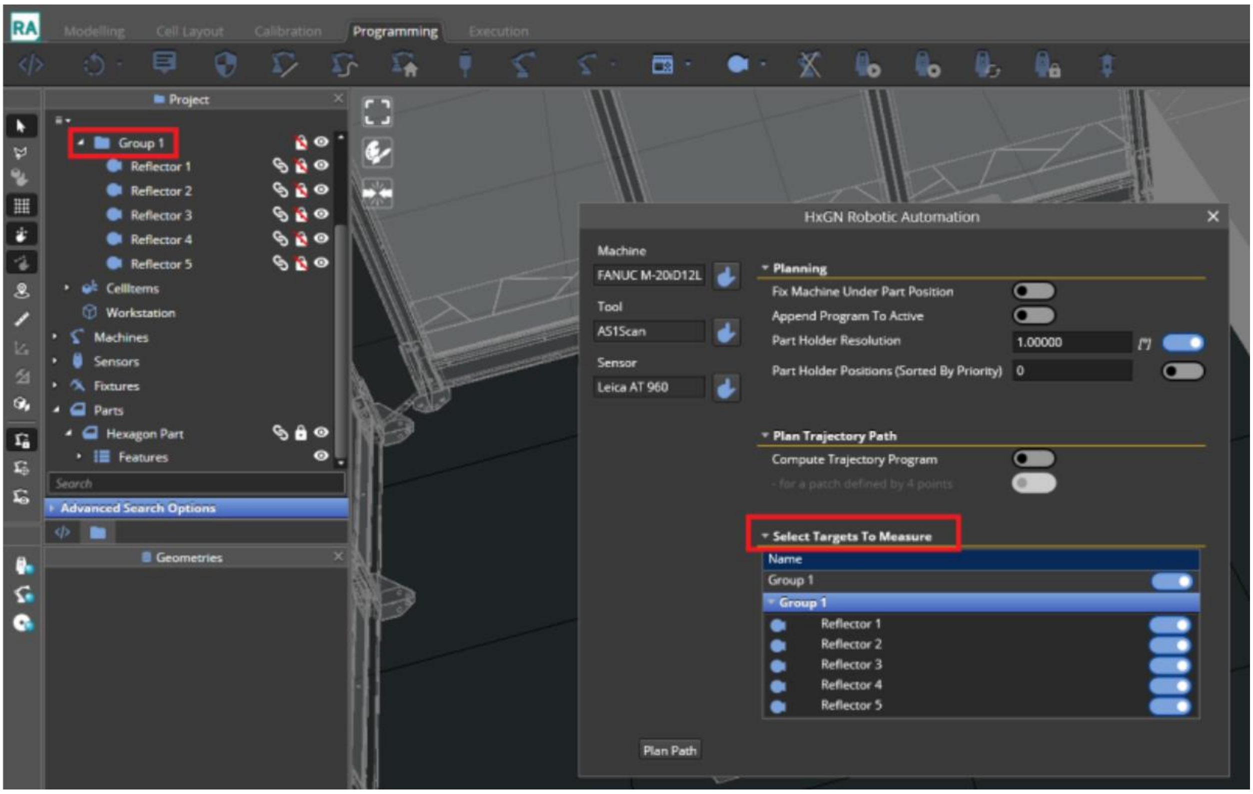
7) Under **Plan Trajectory Path** the user can define two options:

a) **Compute Trajectory Program**: by default, it is checked out. If checked in, the user can select a list of edge points or surface points, and the path planning algorithm will try to find the path to move between these points.

b) **for a patch defined by 4 points**: selecting this mode, the patch scan planning will be used.

Both options refer to the semi-automatic mode programming and will be further explained in chapter 8.9.

8) Under **Select targets to measure**, the user can select the reflector groups to be used for compensation. See chapter 7.2.5 for an explanation on how to create reflector groups.



Machine: FANUC M-20iD12L

Tool: AS1Scan

Sensor: Leica AT 960

Planning

- Fix Machine Under Part Position
- Append Program To Active
- Part Holder Resolution: 1.00000 [°]
- Part Holder Positions (Sorted By Priority): 0

Plan Trajectory Path

- Compute Trajectory Program
- for a patch defined by 4 points

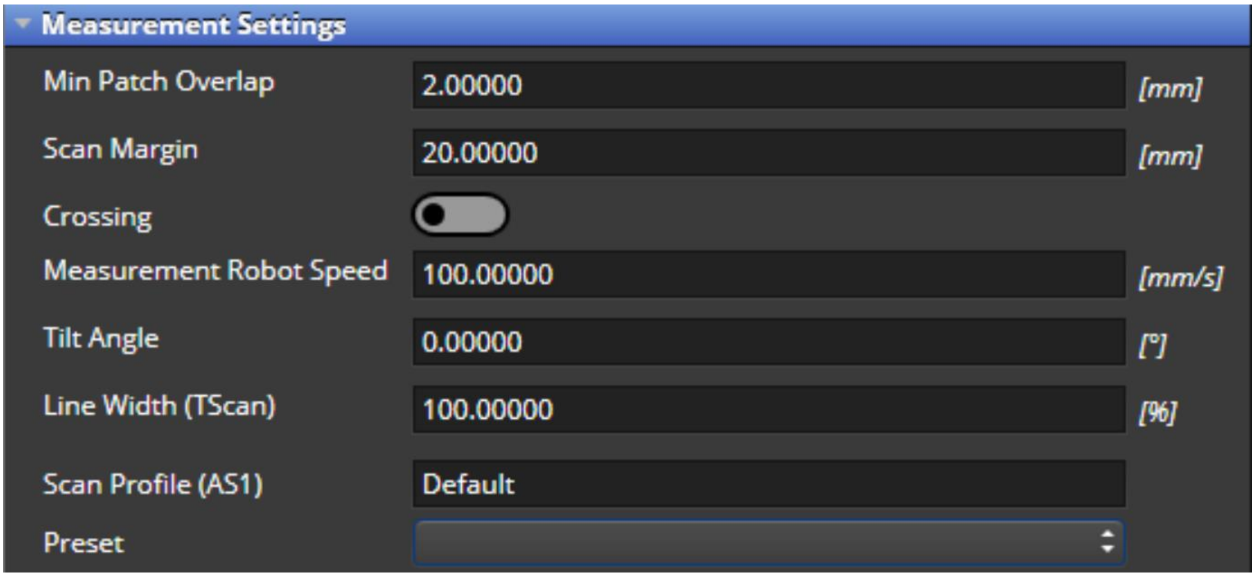
Select Targets To Measure

Name
Group 1
Group 1
Reflector 1
Reflector 2
Reflector 3
Reflector 4
Reflector 5

9) After defining the options for the path, the user can click on the **Plan Path** button and wait for the program to be created. When the program is generated, it will be visible within the Program window. If no solutions is found, the information will be reported in the status window.

8.8.1 Feature measurement settings

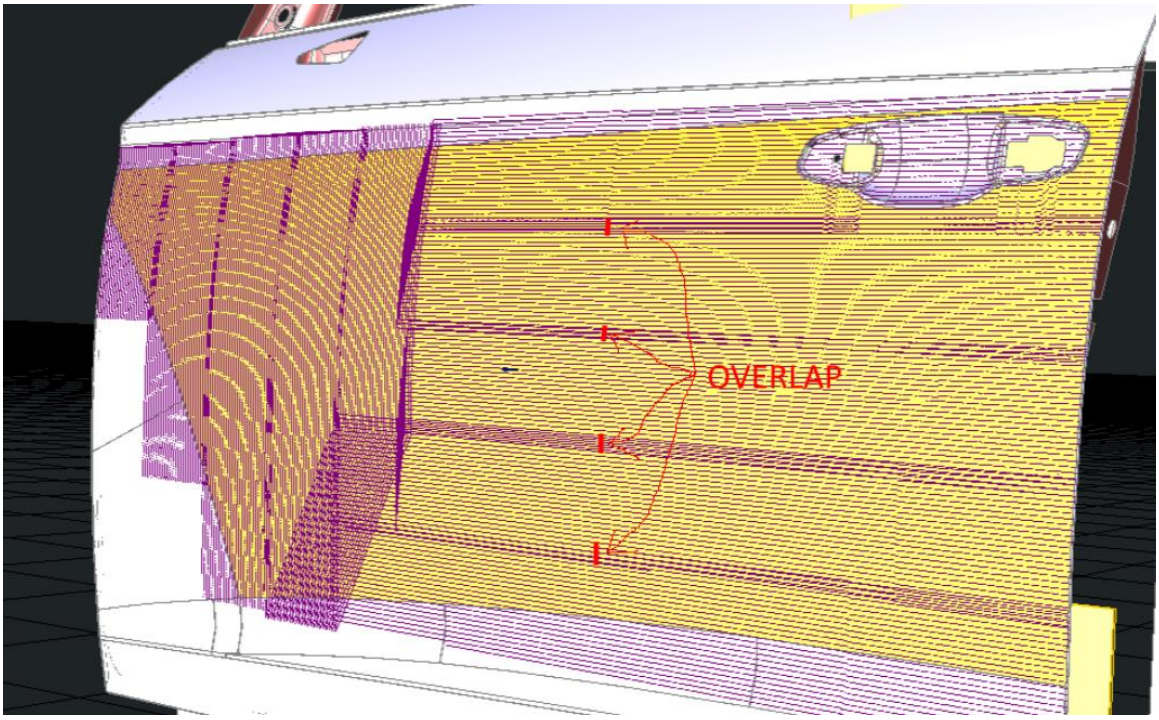
For each feature that is being measured and thus used to compute an automatic path to scan it, it is possible to right-click on it in the project tree and set the following properties:



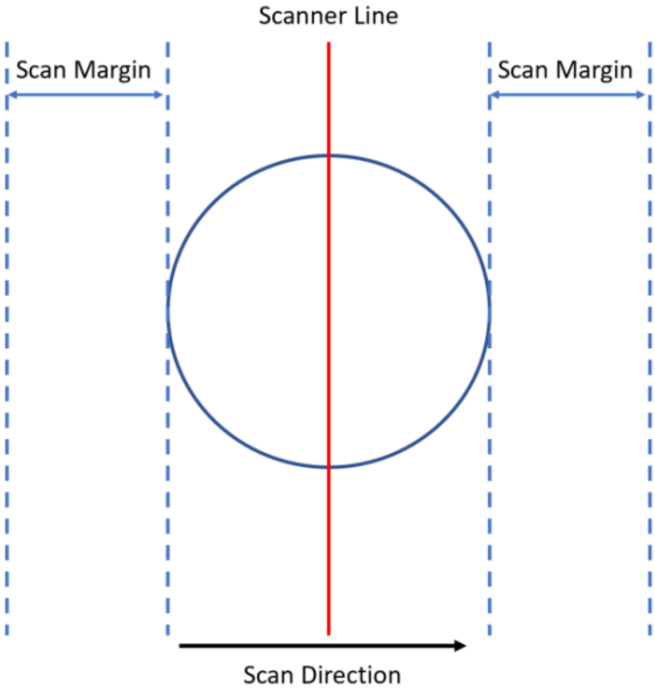
Measurement Settings

Min Patch Overlap	2.00000	[mm]
Scan Margin	20.00000	[mm]
Crossing	○	
Measurement Robot Speed	100.00000	[mm/s]
Tilt Angle	0.00000	[°]
Line Width (TScan)	100.00000	[%]
Scan Profile (AS1)	Default	
Preset		

1) **Min Patch Overlap**: minimum overlap between parallel stripes of a scan of a feature. Note that the overlap can be bigger if more stripes are needed to cover the poly-surface with the specified minimum value.



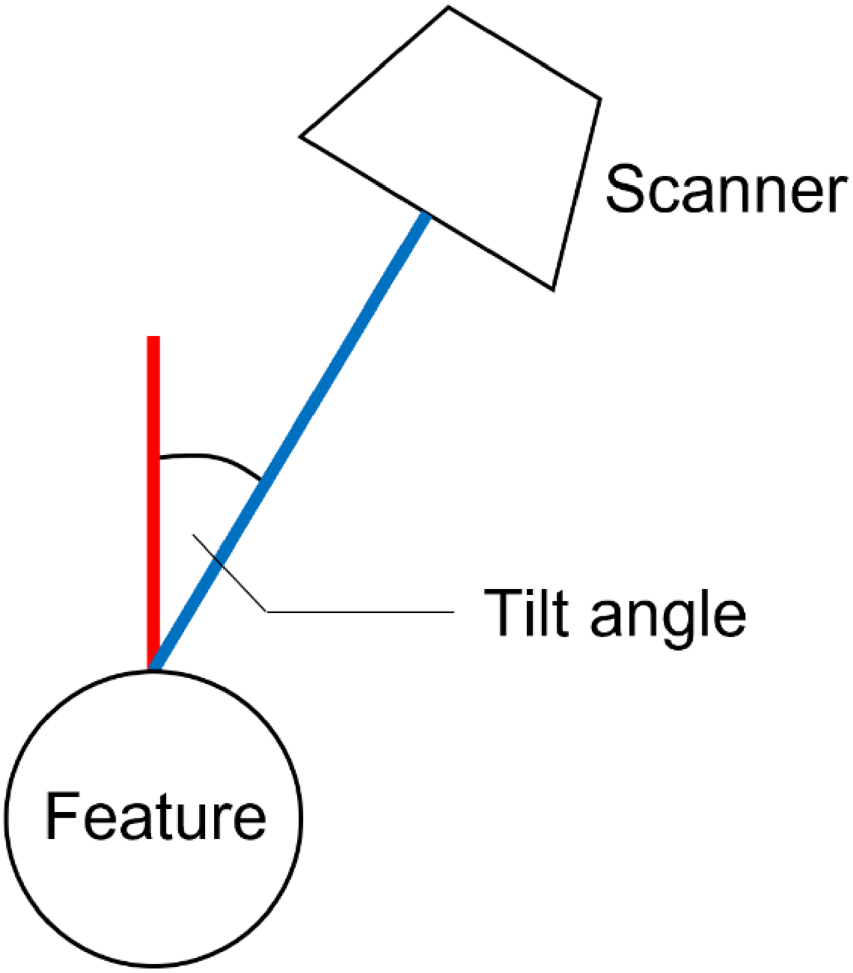
2) **Scan Margin**: the extra margin of the scan around the feature. If it is a point, the scan margin consists of a circle around the point perpendicular to the scan normal within which the scanning will continue to happen. If it is not a point-like feature (e.g. a shape, or a poly surface, or a circle) the margin is to be considered around the perimeter defined by the feature.



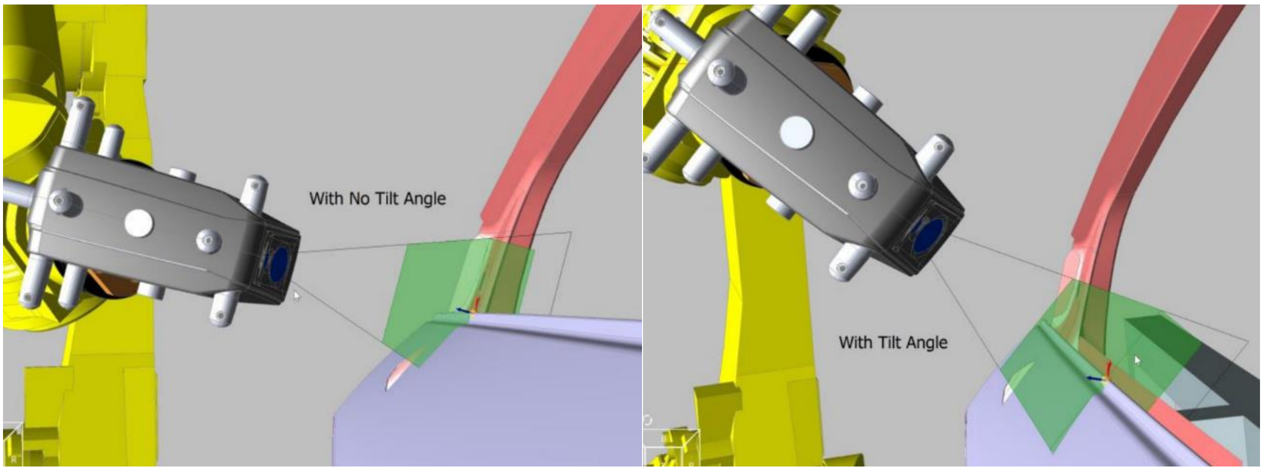
3) **Crossing**: if this is enabled, the feature is first scanned and, if possible, a second scan (the crossing) will be computed considering an angle of 90° ± 45° to use the same scanner face. For the crossing to be considered valid, the same sensor face must be used. If it is not possible to compute the crossing, only the first scan is delivered. No visual keyword in the program informs the user of the success of the crossing. It must be visually verified by the measurement "painting" over the feature.

4) **Measurement Robot Speed**: the speed of the robot during the measurement of the feature, in percentage to the maximum speed:
   a) The default value is 40, meaning that the final robot speed will be set to 40% of its maximum speed.
   b) The user can only change the value to 100, meaning that the final robot speed will be set to 100% of its maximum speed.

5) **Tilt Angle**: it allows the user to specify that the feature is intended to be scanned with a specific tilt along the scan direction as depicted in the picture below:
   a) the red line is the feature normal.
   b) the blue line is the feature normal with the tilt.

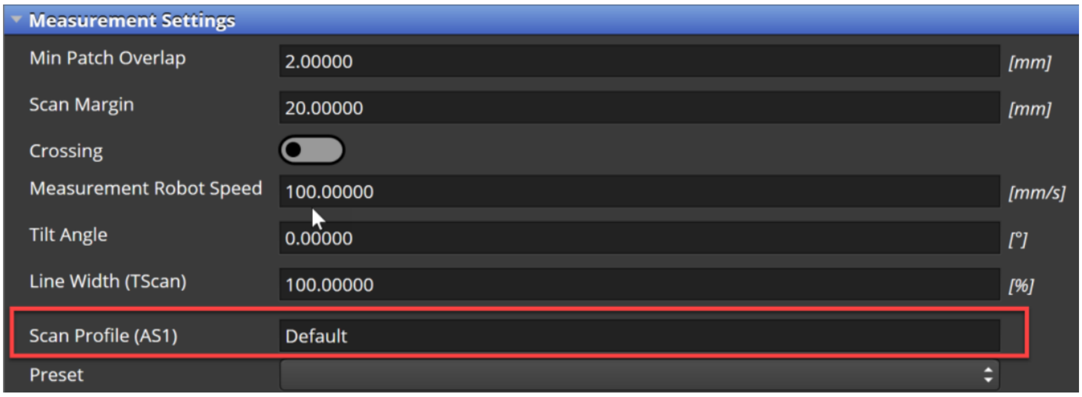


Info: a set of scans is generated over a feature. If the scan with the feature cannot be computed with the tilt angle defined by the user, other tilt values will be tried by the algorithm to provide a solution.



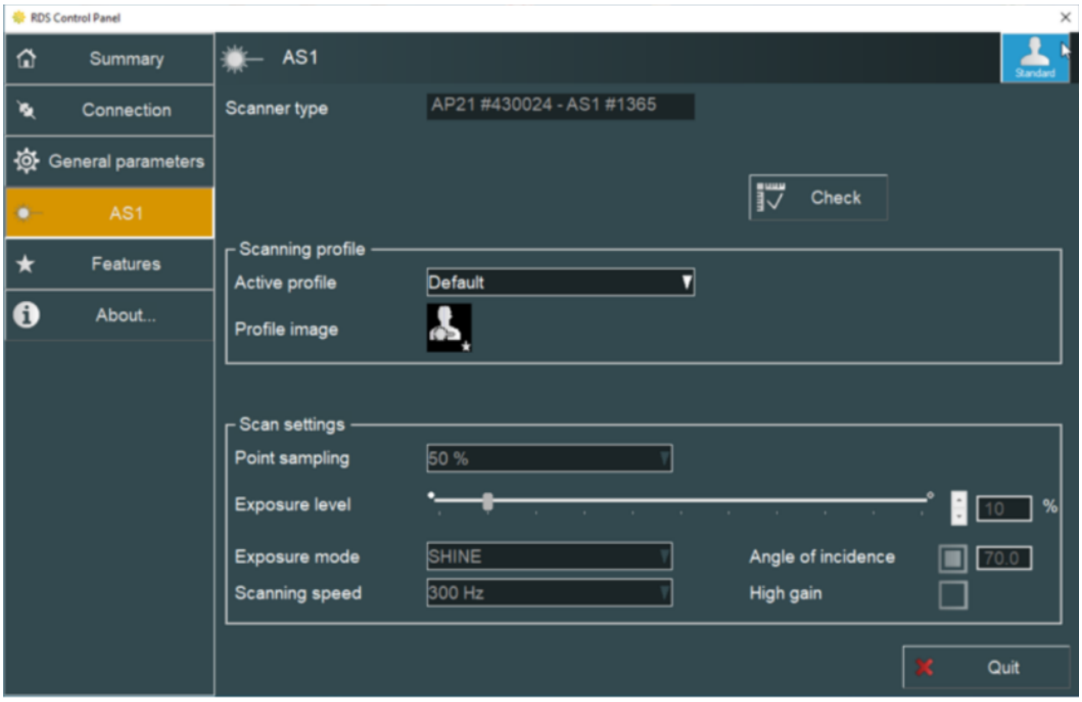
6) **Line Width (T-Scan)**: allows the user to set the T-Scan 5 line width (from 40% to 100%).

7) **Scan Profile (AS1)**: allows the user to activate a Scanning profile defined in RDS.



Measurement Settings

Min Patch Overlap	2.00000	[mm]
Scan Margin	20.00000	[mm]
Crossing	○	
Measurement Robot Speed	100.00000	[mm/s]
Tilt Angle	0.00000	[°]
Line Width (TScan)	100.00000	[%]
Scan Profile (AS1)	Default	
Preset		



AS1

Scanner type: AP21 #430024 - AS1 #1365

Scanning profile

Active profile: Default

Profile image

Scan settings

Point sampling: 50 %

Exposure level: 10 %

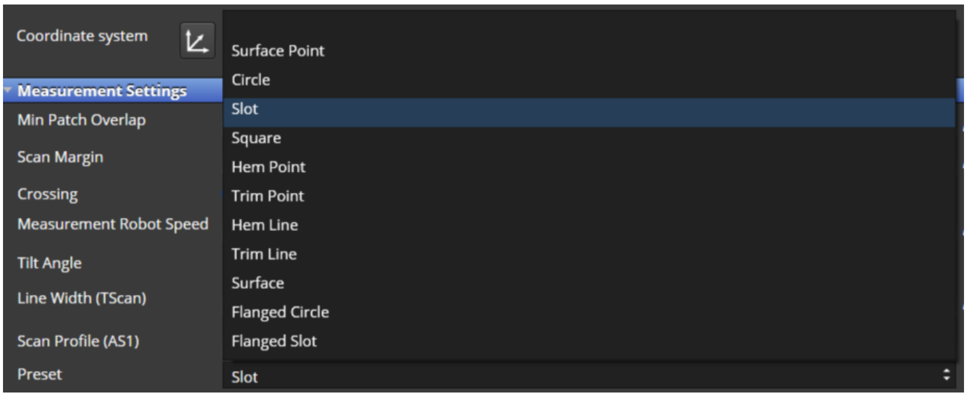
Exposure mode: SHINE

Angle of incidence: 70.0

Scanning speed: 300 Hz

High gain: □

8) **Preset**: allows to specify values for the parameters above, grouped under a specific label that can be defined within the HxGN Robotic Automation settings (see 4.11.1.11).



Coordinate system

Measurement Settings

Min Patch Overlap

Scan Margin

Crossing

Measurement Robot Speed

Tilt Angle

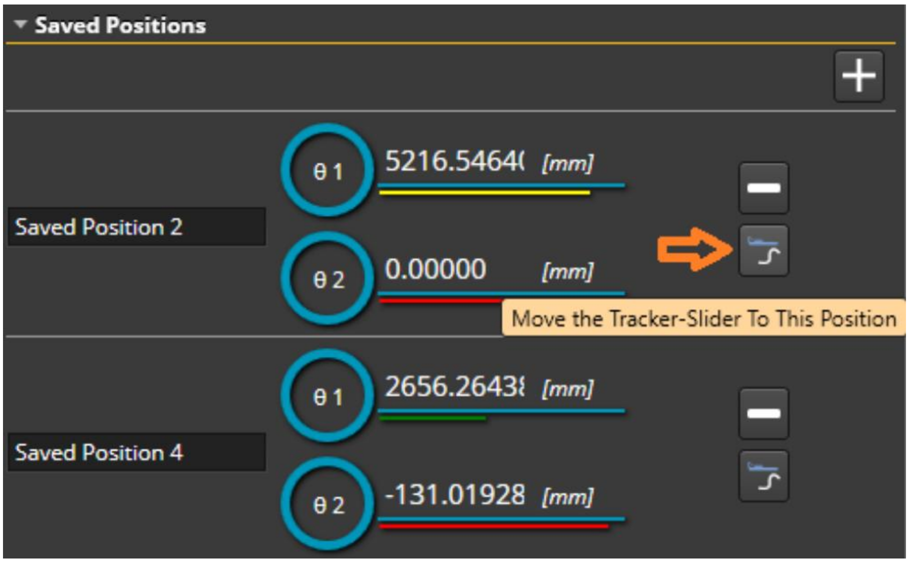
Line Width (TScan)

Scan Profile (AS1)

Preset: Slot

8.8.2 Save fixed positions

It is possible to define fixed position to reduce the range of possible positions of external axes that the automatic path planning will consider creating the robot program.



Saved Positions

Saved Position 2

θ 1: 5216.5464([mm]

θ 2: 0.00000 [mm]

Move the Tracker-Slider To This Position

Saved Position 4

θ 1: 2656.2643! [mm]

θ 2: -131.01928 [mm]

8.8.3 Path planning algorithm flowchart

After the user selects the features and click on plan path after specifying all the required settings as explained in the earlier chapters, the path planning algorithm will create a program according to the flowchart depicted in **Figure 8-4**: Automatic path planning flowchart, that can be summarised in the following steps:

1) Homing motions are added for all the machines in the cell to start the program from the starting positions.

2) Commands to measure the nominal position of the reflectors are added.

3) The command groups to measure the features are created. In case it is not possible to scan the features from the current machines' positions, the machines are moved to a convenient position and the actual positions of the reflectors are measured.

4) After all groups are created and all features are covered, the groups are sorted to supply a best sequence of grouping to have the fastest scan of all the features.

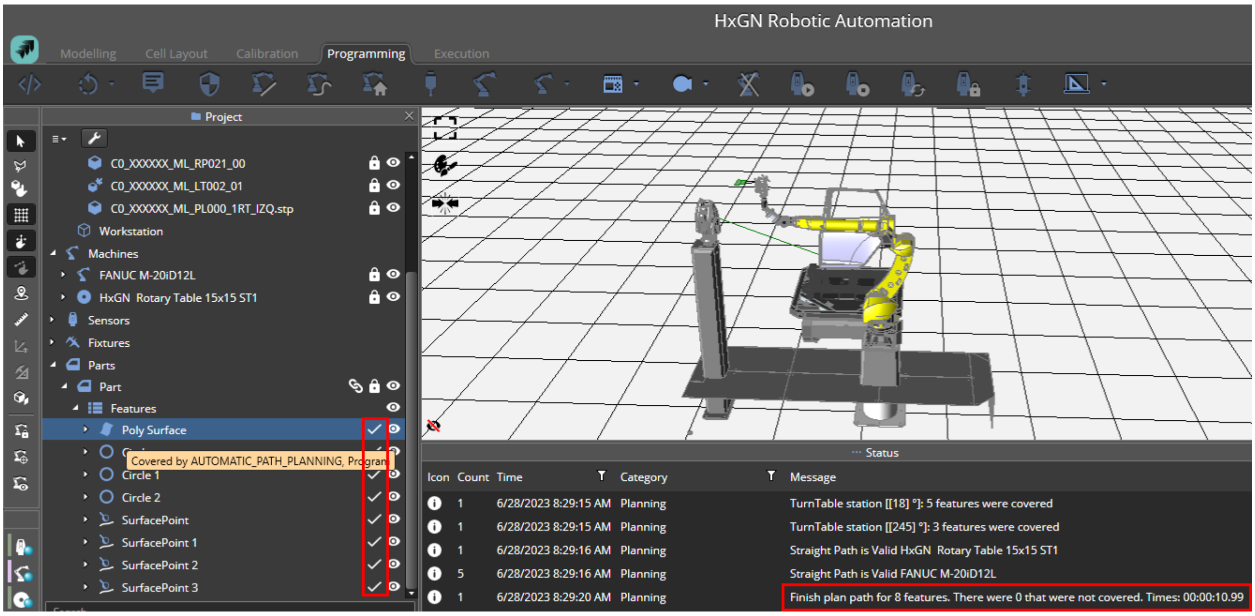
5) If it not possible to move the machines between any groups, more movements are added to allow it.

6) Homing motions are added for all the machines in the cell to end the program correctly by sending the machines back to the starting positions.

7) In case some features cannot be covered by the path planning, the user is informed both in the status window and in the project, tree as explained in section 8.8.4.

8.8.4 Evaluate the quality of the scan

After the program generated by the automatic path planning is validated and executed, it is possible to evaluate if the features were properly covered, i.e., if enough points were collected during the scan of the features. The automatic path planning algorithm tries to scan a feature in the best way. In case the features are scanned properly and covered properly, a checkmark icon is displayed near the feature.



- C0_XXXXXX_ML_RP021_00
- C0_XXXXXX_ML_LT002_01
- C0_XXXXXX_ML_PL000_1RT_IZQ.stp
- Workstation
- Machines
  - FANUC M-20iD12L
  - HxGN Rotary Table 15x15 ST1
- Sensors
- Fixtures
- Parts
  - Part
    - Features
      - Poly Surface ✓
      - C… ✓ Covered by AUTOMATIC_PATH_PLANNING, Program
      - Circle 1 ✓
      - Circle 2 ✓
      - SurfacePoint ✓
      - SurfacePoint 1 ✓
      - SurfacePoint 2 ✓
      - SurfacePoint 3 ✓

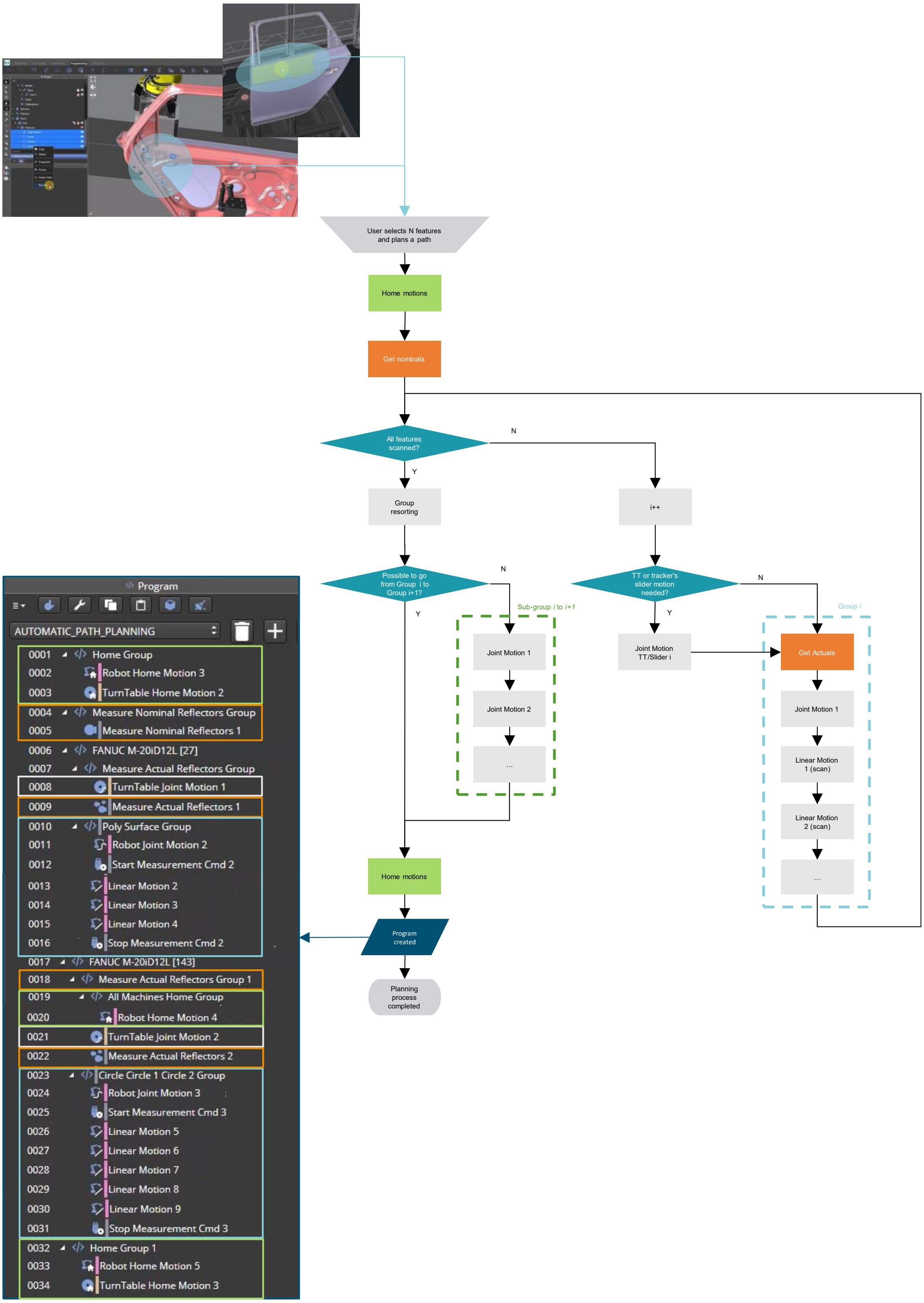
Status

```
Icon Count Time                  Category   Message
 ⓘ    1    6/28/2023 8:29:15 AM  Planning   TurnTable station [[18] °]: 5 features were covered
 ⓘ    1    6/28/2023 8:29:15 AM  Planning   TurnTable station [[245] °]: 3 features were covered
 ⓘ    1    6/28/2023 8:29:16 AM  Planning   Straight Path is Valid HxGN  Rotary Table 15x15 ST1
 ⓘ    5    6/28/2023 8:29:16 AM  Planning   Straight Path is Valid FANUC M-20iD12L
 ⓘ    1    6/28/2023 8:29:20 AM  Planning   Finish plan path for 8 features. There were 0 that were not covered. Times: 00:00:10.99
```

If this is not possible to cover the feature, the automatic path planning tries to scan it with lower-quality scans. This situation can happen e.g., measuring edge points that could be scanned not with the highest quality.

In this case the user can decide to adjust the scans to cover the partially covered feature or try to execute the feature computation in the metrology SW with the acquired data.

Info: the concept of coverage is not to be confused with a metrology computation on the features and provides only and indication on COP quality for a specific feature. Any metrological analysis must be done in the metrology software used in the project (see chapter 0).



```
0001  Home Group
0002    Robot Home Motion 3
0003    TurnTable Home Motion 2
0004  Measure Nominal Reflectors Group
0005    Measure Nominal Reflectors 1
0006  FANUC M-20iD12L [27]
0007    Measure Actual Reflectors Group
0008      TurnTable Joint Motion 1
0009      Measure Actual Reflectors 1
0010    Poly Surface Group
0011      Robot Joint Motion 2
0012      Start Measurement Cmd 2
0013      Linear Motion 2
0014      Linear Motion 3
0015      Linear Motion 4
0016      Stop Measurement Cmd 2
0017  FANUC M-20iD12L [143]
0018    Measure Actual Reflectors Group 1
0019      All Machines Home Group
0020        Robot Home Motion 4
0021      TurnTable Joint Motion 2
0022      Measure Actual Reflectors 2
0023    Circle Circle 1 Circle 2 Group
0024      Robot Joint Motion 3
0025      Start Measurement Cmd 3
0026      Linear Motion 5
0027      Linear Motion 6
0028      Linear Motion 7
0029      Linear Motion 8
0030      Linear Motion 9
0031      Stop Measurement Cmd 3
0032  Home Group 1
0033    Robot Home Motion 5
0034    TurnTable Home Motion 3
```

***Figure 8-4**: Automatic path planning flowchart*

8.8.5 Features supported by Path Planning and relevant parameters

The following table supplies a quick reference on which features are supported by the path planning algorithm and which parameters can be changed to obtain different results. For more details on how to change the parameters, please refer to Hexagon Metrology Devices - AS1 Best Practice and Hexagon Metrology Devices - T-Scan5 Best Practice. For a list of features that can be imported from metrology applications please refer to chapter 0.

Feature type	Supported	Preferent Scan Direction (1)	Crossing	Overlap	Scanned as poly-surface
Circle	✓	✗	✓	✗	✓
Cone	✗	-	-	-	✗
Cylinder	✓	✗	✗	✗	✗
Edge Line	✗	-	-	-	✗
Edge Point	✗	✗	✗	✗	✗
Ellipse	✓	✗	✓	✗	✗
Flanged Circle	✗	-	-	-	✗
Flanged Slot	✗	-	-	-	✗
Hem Point	✗	-	-	-	✗
Hexagon	✗	-	-	-	✗
Plane	✓	✗	✗	✓	✓
Poly-surface	✓	✗	✗	✓	✗
Polyline	✓(2)	✓	✗	✗	✓
Slot	✓	✓	✓	✗	✗
Sphere	✓	✗	✗	✗	✗
Surface Point	✓	✗	✓	✗	✓
Key Hole	✗	-	-	-	✗

(1) Some features have a specific direction in which they are scanned (e.g. a slot is scanned diagonally and not following the major axis direction)

(2) An extra step to deconstruct the polyline into a set of edge points is needed before the path planning algorithm can be used on it.

8.9 Semi-automatic mode programming [from V2.0.1]

The semi-automatic approach is called in this way because it provides a simplified way of creating the measurement part of the program including joint and linear motions, homing motions and start/stop measurement commands to execute the scanning of an area of interest identified on the part.

The area of interest can be defined in two ways:

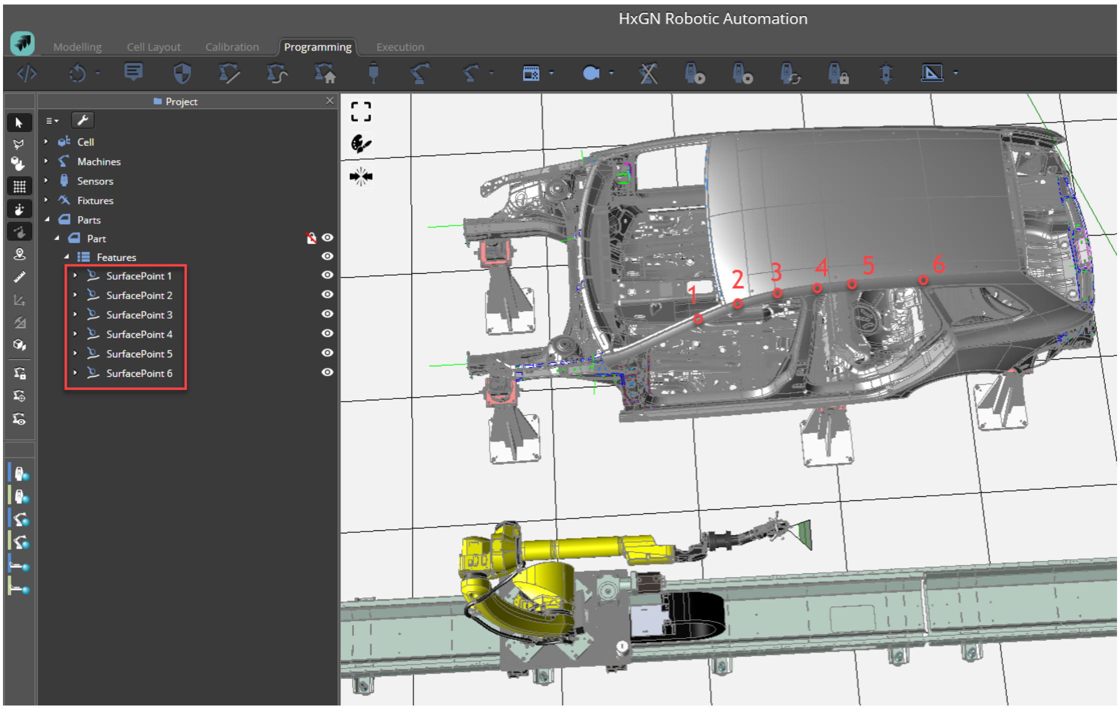
1) by a list of points to be followed by the scanner to perform the measurement along the defined piecewise trajectory as described in chapter 8.9.1

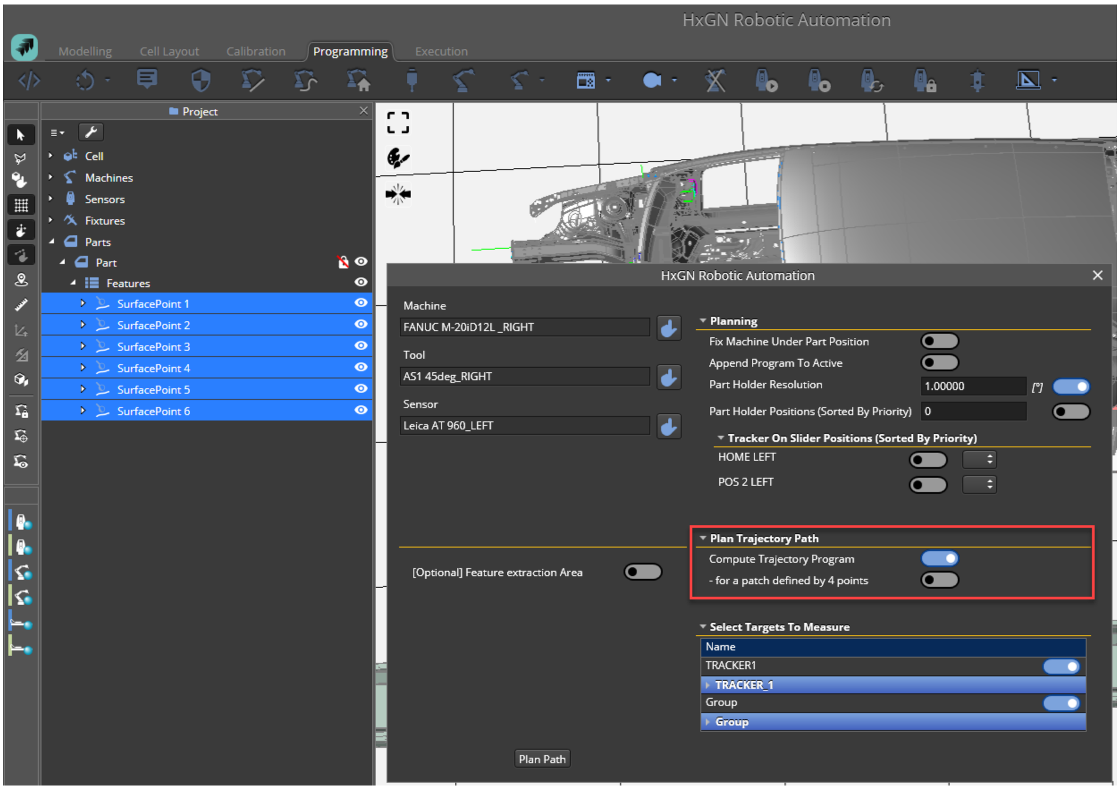
2) by a set of four points defining a four-sided polygon within which a scanning trajectory will be computed in order to cover the defined area as described in chapter 8.9.2

Unlike the automatic mode programming, the user still needs to insert compensation commands manually to compensate for the relationship change between tracker and part.

8.9.1 Trajectory scanning

The same procedure described for the automatic path planning mode can be followed but the features in this this case are defined by the user by inserting surface points with the feature generation mode (see Chapter 6.2) and planning a path on those points by selecting them from the project tree, right click and then select the **Compute Trajectory Program** option within the Planning dialog box as depicted in the image below.





HxGN Robotic Automation

Machine: FANUC M-20iD12L_RIGHT

Tool: AS1 45deg_RIGHT

Sensor: Leica AT 960_LEFT

[Optional] Feature extraction Area

Planning
- Fix Machine Under Part Position
- Append Program To Active
- Part Holder Resolution: 1.00000 [°]
- Part Holder Positions (Sorted By Priority): 0

Tracker On Slider Positions (Sorted By Priority)
- HOME LEFT
- POS 2 LEFT

Plan Trajectory Path
- Compute Trajectory Program
- - for a patch defined by 4 points

Select Targets To Measure

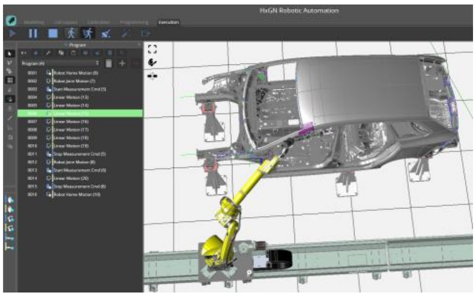
Name
TRACKER1
TRACKER_1
Group
Group

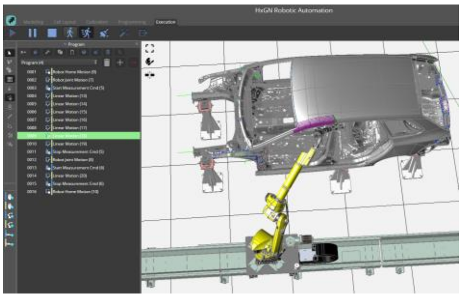
Plan Path

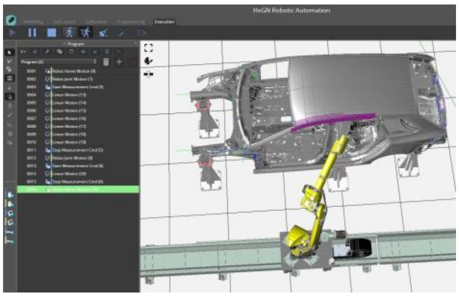
After clicking on **Plan Path** button, it is just needed to wait for the program to be created as for automatic mode programming. When the program is generated, it will be visible within the Program window. If no solutions is found, the information will be reported in the status window.

The path planning algorithm will generate a measurement program containing joint and linear motions to scan the area defined by the four surface points as well as homing motions and start/stop measurement commands to execute the scanning.

In the following pictures is shown how a trajectory scanning program will look like for the list of six surface points defined above.

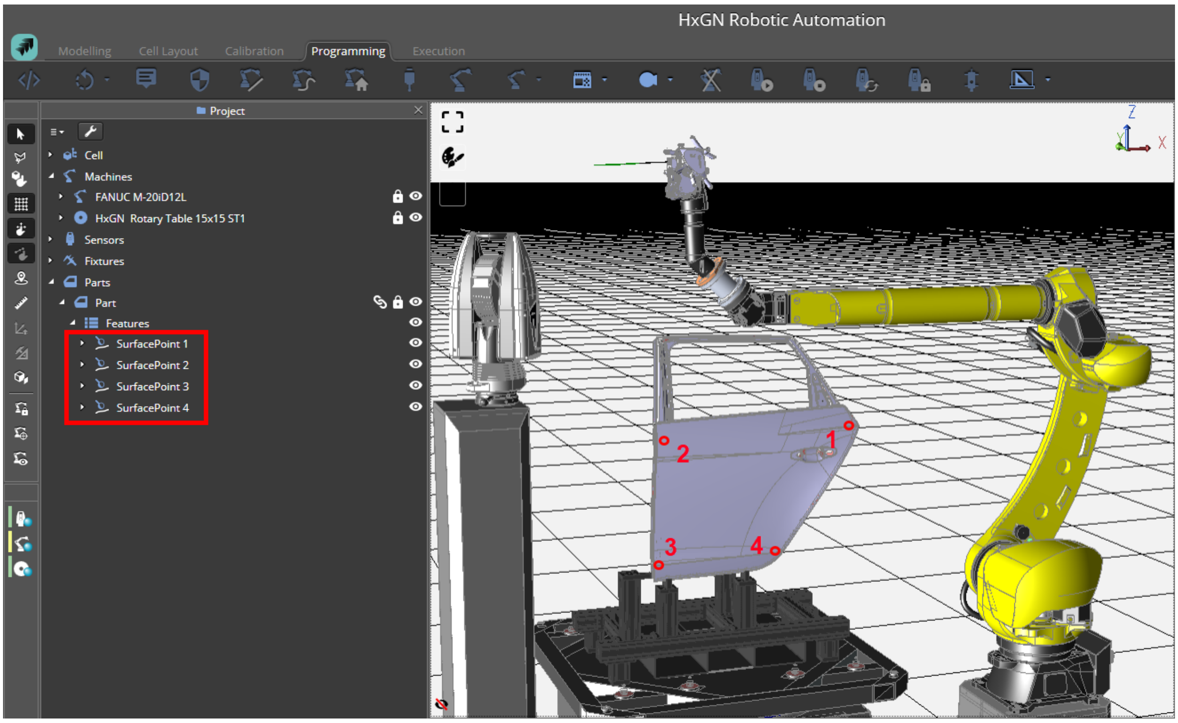




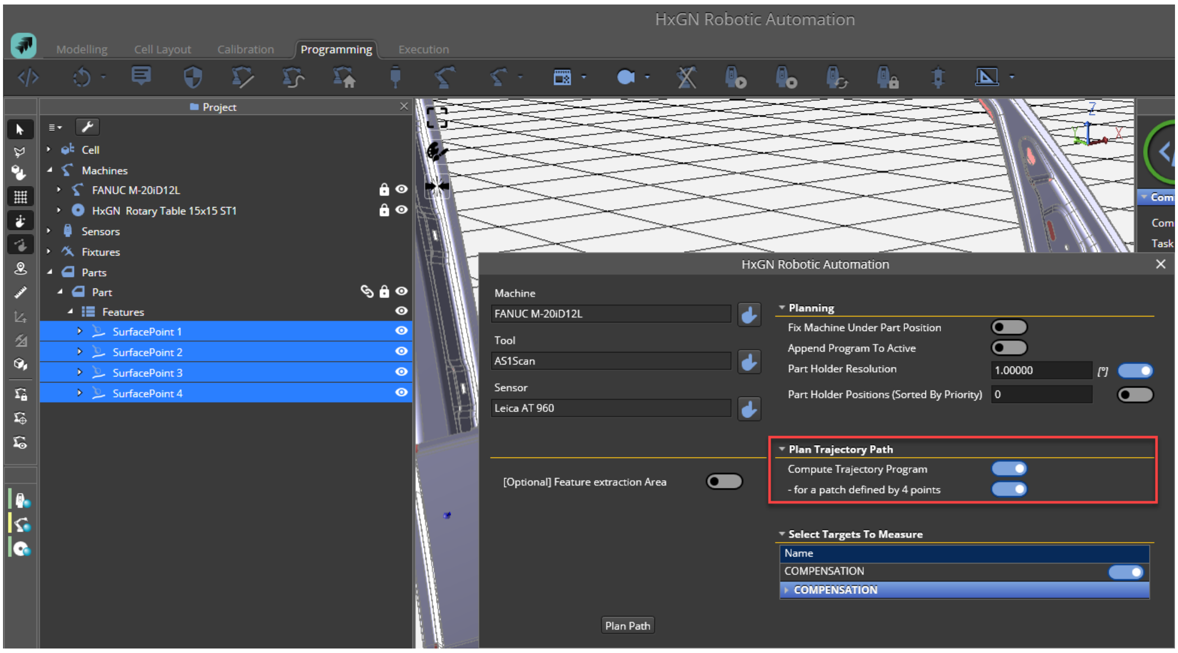


8.9.2 Patch scanning

This programming mode allows HxGN Robotic Automation to create a trajectory path planning to scan a surface defined by four surface point features.



The same procedure described in the previous chapter can be followed but the features in this this case are defined by the user by inserting surface points defining a four-sided polygonal area and generating a program to scan the area by selecting the points from the project tree, right click and then select the **Compute Trajectory Program** option (as done in the previous chapter) but in this case also checking **– for a patch defined by 4 points** within the Planning dialog box as depicted in the image below.



Machine: FANUC M-20iD12L
Tool: AS1Scan
Sensor: Leica AT 960
[Optional] Feature extraction Area: off

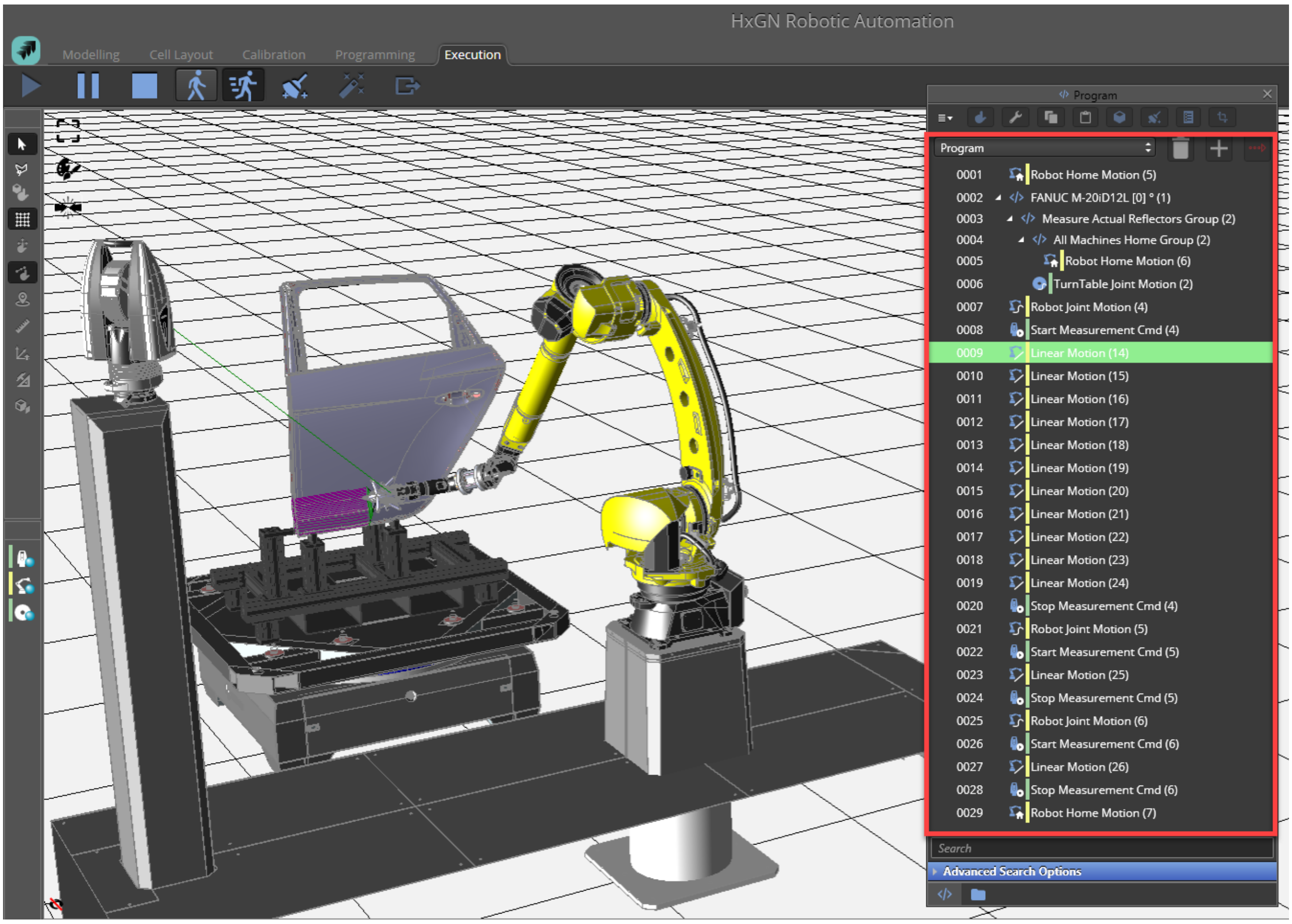
Planning
Fix Machine Under Part Position: off
Append Program To Active: off
Part Holder Resolution: 1.00000 [°] on
Part Holder Positions (Sorted By Priority): 0 off

Plan Trajectory Path
Compute Trajectory Program: on
- for a patch defined by 4 points: on

Select Targets To Measure

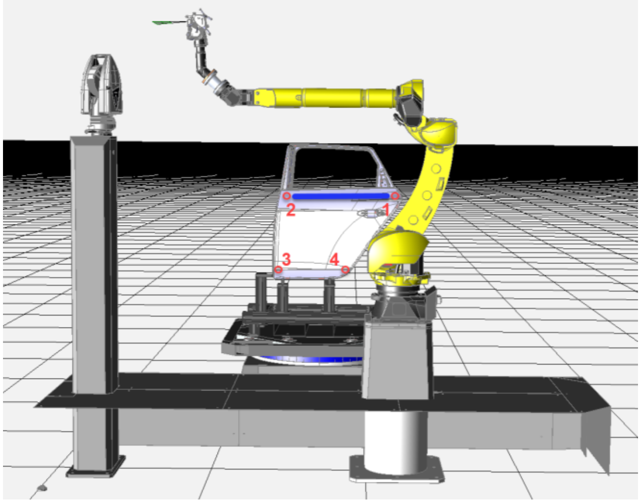
Name	
COMPENSATION	on
COMPENSATION	

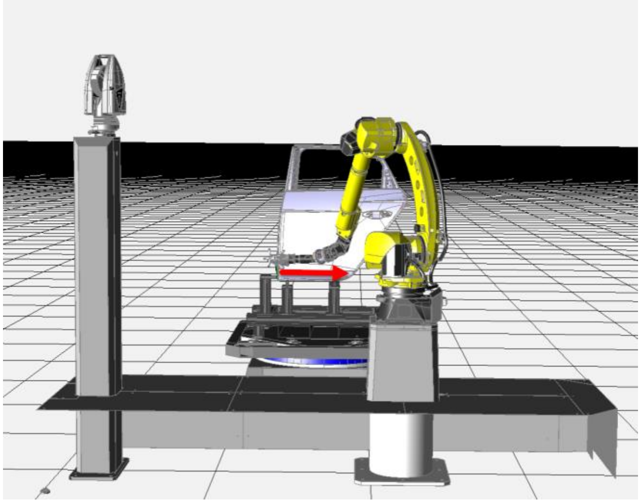
The path planning algorithm will generate a measurement program containing joint and linear motions to scan the area defined by the four surface points as well as homing motions and start/stop measurement commands to execute the scanning.



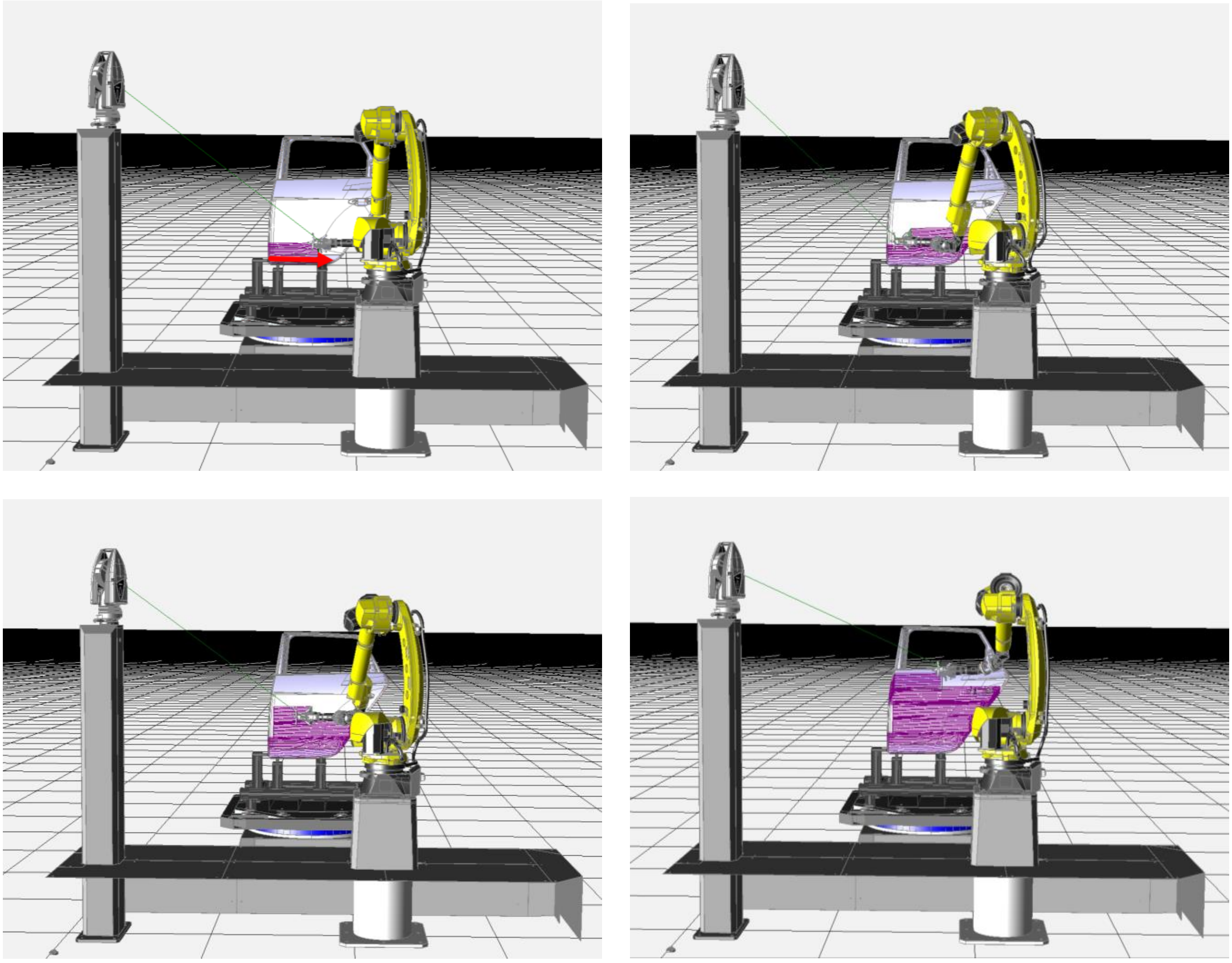
```
Program
0001      Robot Home Motion (5)
0002  ◢ </> FANUC M-20iD12L [0] ° (1)
0003    ◢ </> Measure Actual Reflectors Group (2)
0004      ◢ </> All Machines Home Group (2)
0005            Robot Home Motion (6)
0006          TurnTable Joint Motion (2)
0007      Robot Joint Motion (4)
0008      Start Measurement Cmd (4)
0009      Linear Motion (14)
0010      Linear Motion (15)
0011      Linear Motion (16)
0012      Linear Motion (17)
0013      Linear Motion (18)
0014      Linear Motion (19)
0015      Linear Motion (20)
0016      Linear Motion (21)
0017      Linear Motion (22)
0018      Linear Motion (23)
0019      Linear Motion (24)
0020      Stop Measurement Cmd (4)
0021      Robot Joint Motion (5)
0022      Start Measurement Cmd (5)
0023      Linear Motion (25)
0024      Stop Measurement Cmd (5)
0025      Robot Joint Motion (6)
0026      Start Measurement Cmd (6)
0027      Linear Motion (26)
0028      Stop Measurement Cmd (6)
0029      Robot Home Motion (7)
```

The user must be aware that the scan trajectory orientation is determined by the line defined by the first two points (e.g. “SurfacePoint 1” and “SurfacePoint 2” as shown in the example below, coloured in blue).
The starting point and the starting direction of the generated program is determined by the algorithm in order to start from the closest point to the robot starting position (e.g. from “SurfacePoint 3” in the example below).

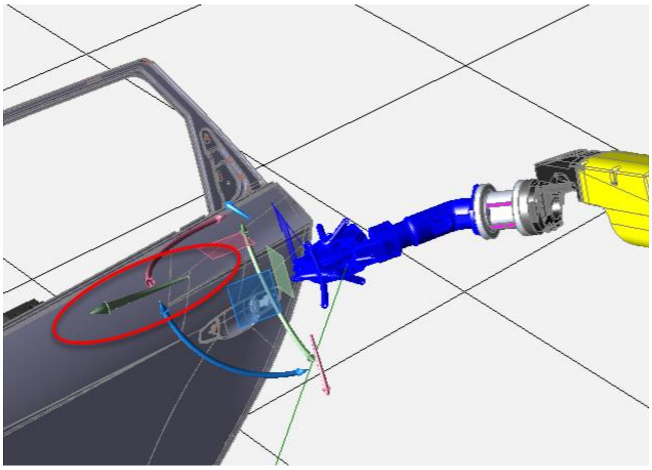


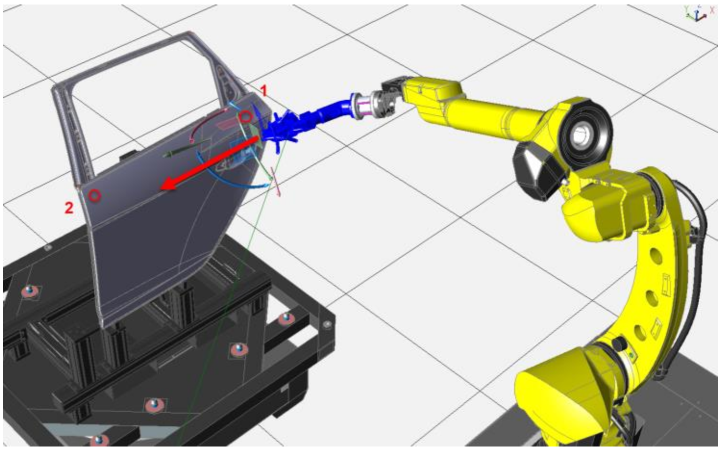


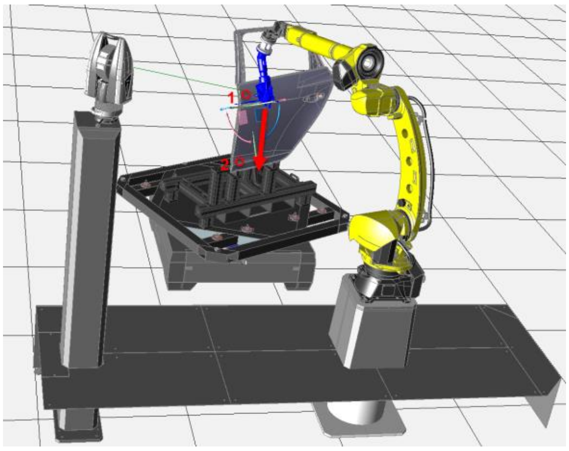
In the following pictures is shown how a patch scanning program will look like for the list of four surface points defined above.



The tool orientation is also determined by the first two points as the direction in which the tool will be oriented to perform the scan on the positive Y direction of the TCP. This is also valid for the trajectory scanning described in the previous chapter.







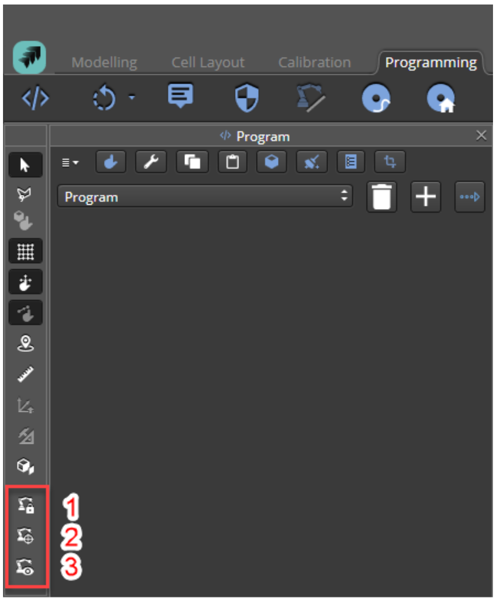
Pro Tip: This approach is a clever way to exploit the high point density of the AS1 scanner in a “featureless” approach where the features are not considered explicitly as in the automatic path planning but rather covered implicitly by having patch scanning which is agnostic on the features contained in the area and will result in the fastest execution time.

8.10 Manual mode programming

In manual mode, the user would need to define the position where to move the robot to start a scan and the linear movements needed to scan the feature or area of interest. It is also needed to add all necessary compensation and homing commands to create a valid inspection program.
To do it, HxGN Robotic Automation have different modes that the user can choose depending on their needs.

8.10.1 Programming mode special toolbar commands

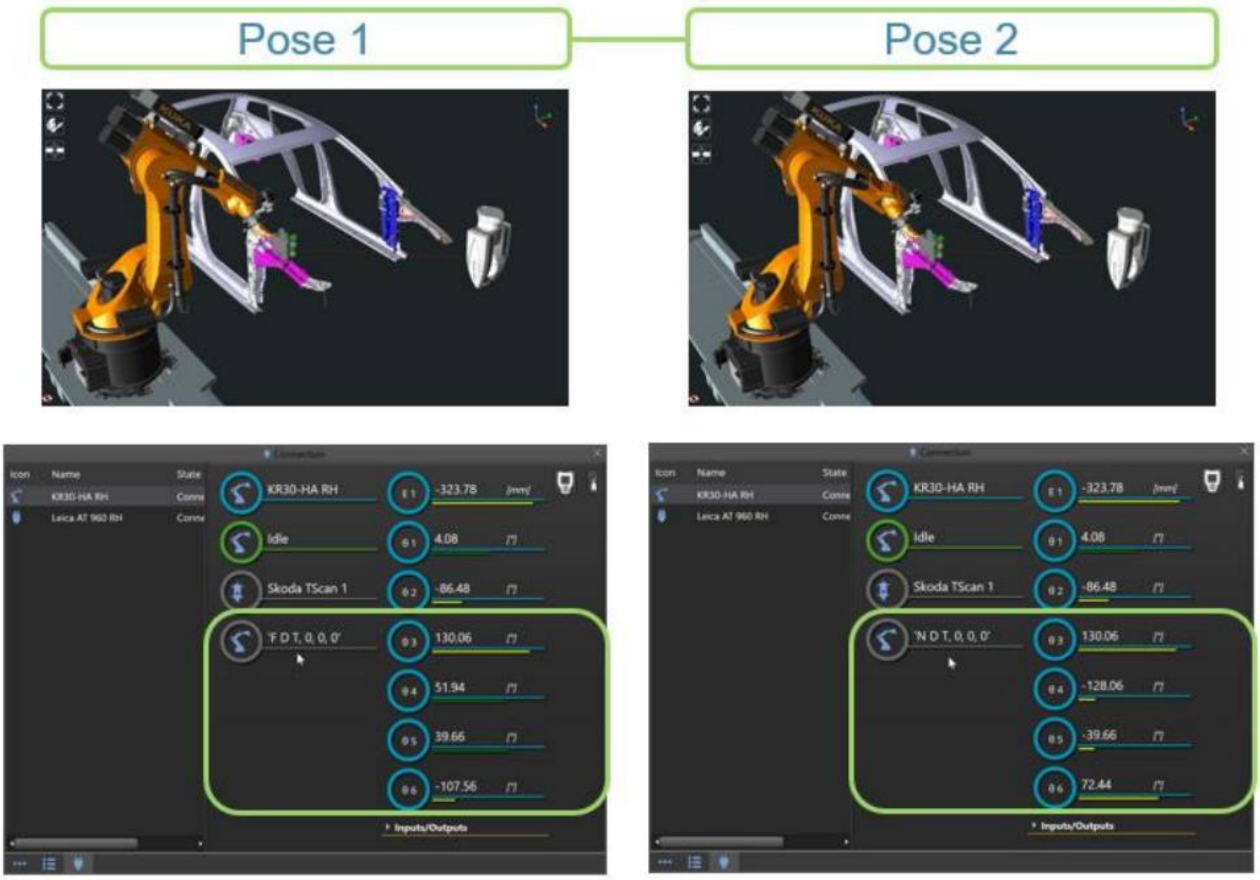
When programming mode is selected the special modes toolbar will be expanded with three more commands.



1) **Lock Machines**: when enabled, allows coupled machines to be moved together, if it is disabled, those machines will be moved independently.

2) **Hold Cartesian Position**: when enabled, robot will move without changing the position of the TCP in the cartesian space, if it is disabled the robot will move with no restrictions.

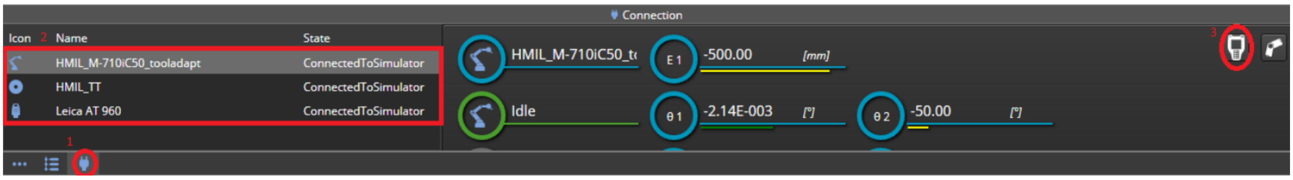
3) **Change Machine Pose**: option to switch from the Multiple Robot Valid Poses (different Inverse Kinematics result) to reach a specific target TCP position. Useful when doing Manual Programming or program fine tuning to avoid Robot Configuration changes or singularities.



8.10.2 Controlling a cell item using the teach pendant

All machines that can be moved by the user have their own virtual teach pendants. The user can move them using the different settings available. To activate a teach pendant:

1) Select the **Connection** tab on the bottom window.
2) Choose the relevant machine.
3) Click on the **Teach Pendant** icon and the teach pendant will open.


Connection

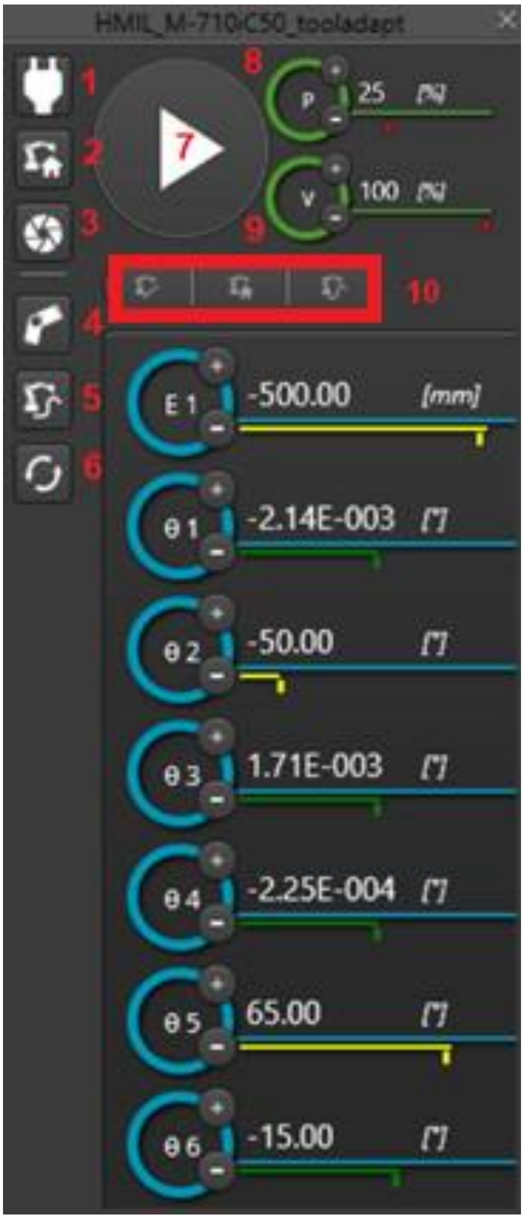
Icon	Name	State
	HMIL_M-710iC50_tooladapt	ConnectedToSimulator
	HMIL_TT	ConnectedToSimulator
	Leica AT 960	ConnectedToSimulator

HMIL_M-710iC50_t
Idle
E1 -500.00 [mm]
θ1 -2.14E-003 [°]
θ2 -50.00 [°]

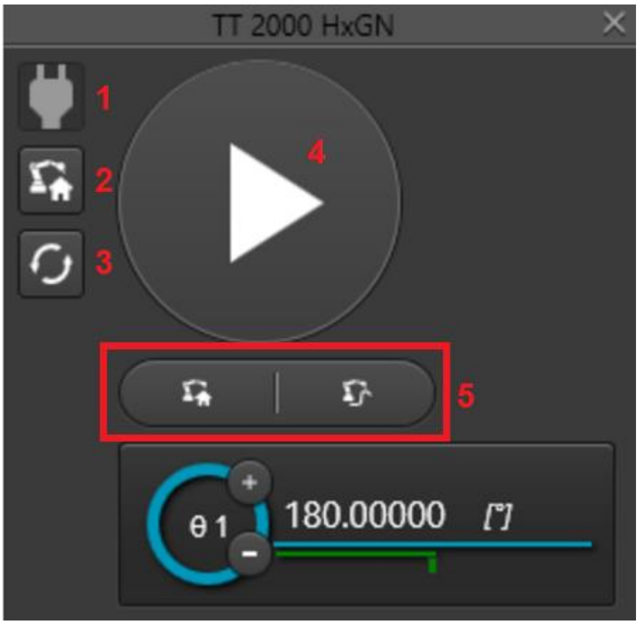
8.10.2.1 Robot teach pendant

The following settings are available:

1) **Connect/Disconnect**: connects or disconnects from the real hardware.
2) **Go Home**: takes the robot to home position.
3) **Show GoPro**: shows the camera of the tool (if one is attached to the machine).
4) **Position**: switches between different robot coordinate systems (Joint, World, Tool) to show the relative values accordingly.
5) **Motion Mode**: selects the way the robot will move (joint or linear motions).
6) **Refresh**: updates the robot position to fit the 3D environment.
7) **Play/Stop**: moves/stops robot.
8) **Program Override**: the percentage of the maximum velocity used.
9) **Velocity**: maximum speed of the robot.
10) Buttons to add **linear**, **home**, or **joint motion** commands to the active program.

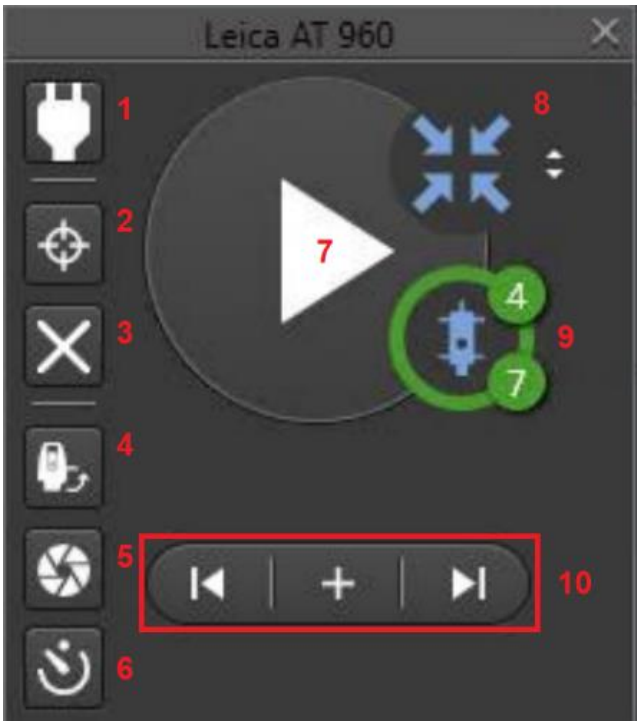


8.10.2.2 Turn table/slider teach pendant



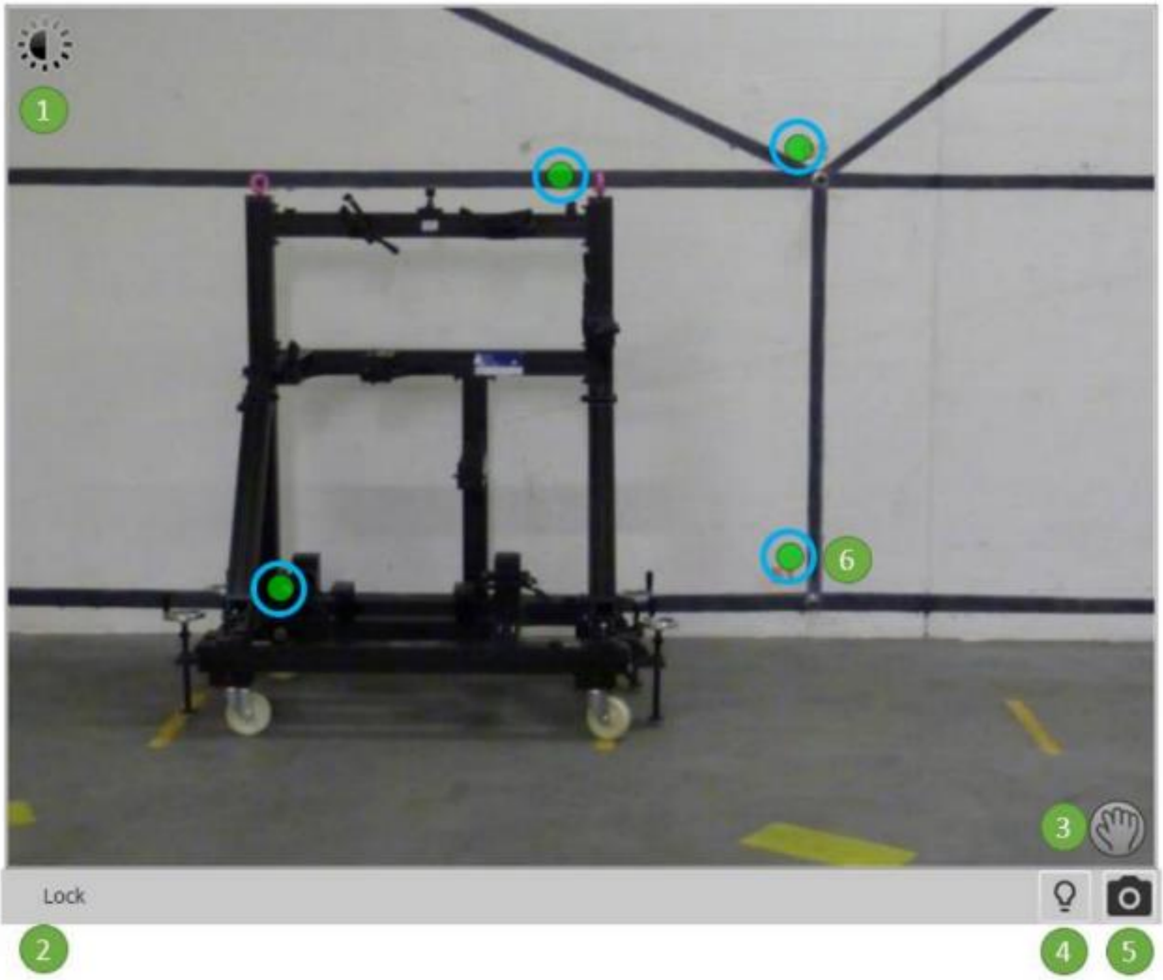
The following settings are available for external axes controlled by the PLC like a turn table or a linear slider:
1) **Connect/Disconnect**: connects or disconnects from the real hardware.
2) **Go Home**: takes the machine to home position.
3) **Refresh**: updates the machine position to fit the 3D environment.
4) **Play/Stop**: moves/stops the machine.
5) Buttons to add **home** or **joint motion** commands to the active program.

8.10.2.3 Tracker teach pendant

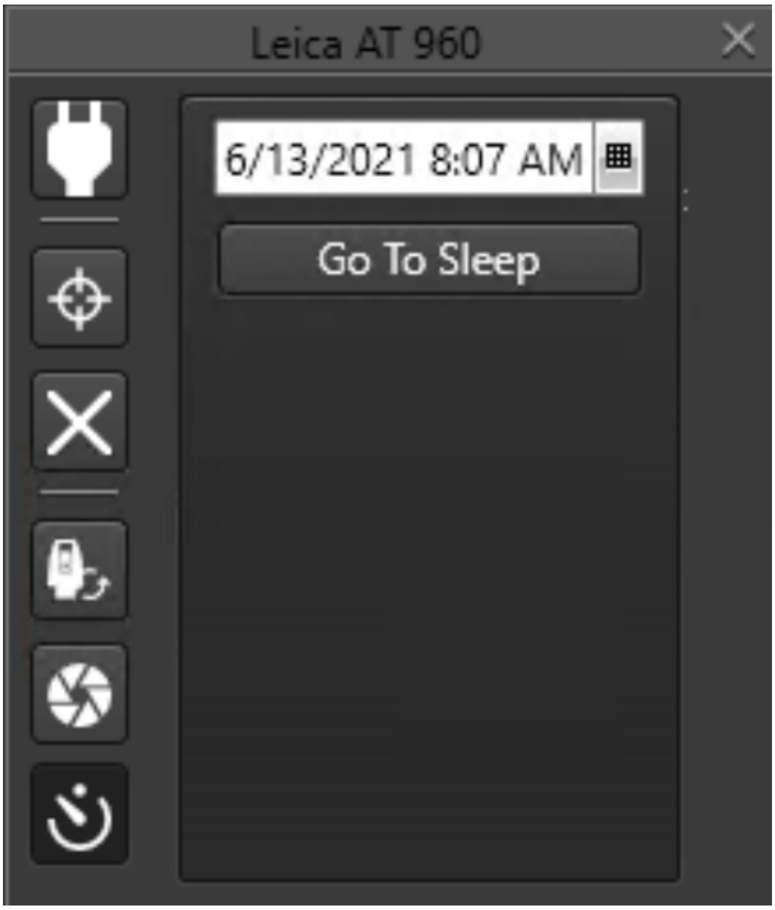


The following settings are available:

1) **Connect/Disconnect**: connects or disconnects to the tracker.
2) **Lock on**: locks on target tool, e.g., AS1 scanner or a reflector.
3) **Delete last point**: removes last point measured.
4) **Initialize**: initializes the tracker.
5) **Show GoPro**: click to open OVC of tracker.
   User can move tracker joints with the manipulator.
   a) Brightness: user can change brightness
   b) Enable/disable Lock: enable to lock on target /disable to lock on target.
   c) Manipulator: use to mode joints of tracker
   d) Help button: click to get help on commands.
   e) Take snapshot: click to take a snapshot. Immediately a folder window will appear to save it.
   f) Targets will glow in green as seen on the image below.



6) **Turn Off**: turns off the laser and set the wake-up time. To be used to set the wake-up time by clicking on the date and time field and put tracker to sleep by clicking on the Go To Sleep button.

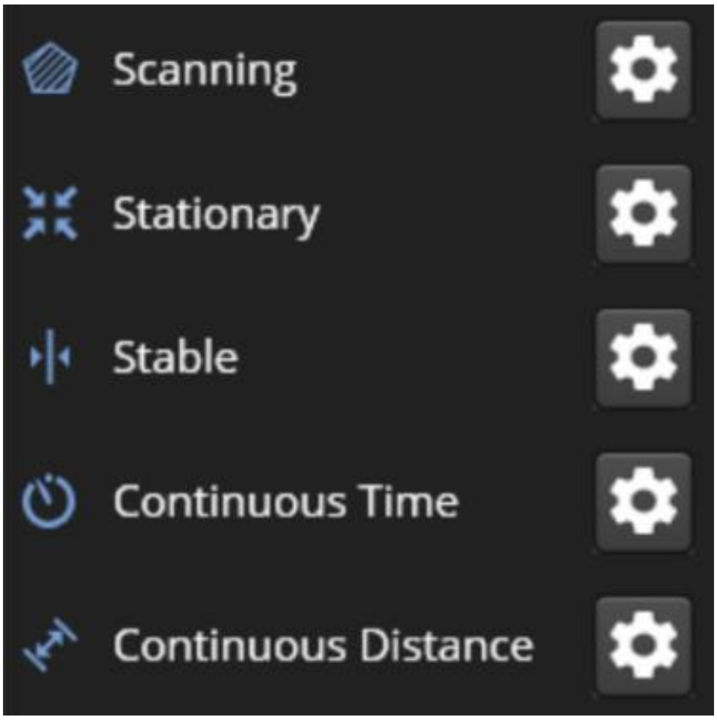


6/13/2021 8:07 AM

7) **Measure/Stop**: starts/stops measurement.
8) **Measure mode**: click on four arrows to access measurement mode options.
   User can select one of the five modes available from the dropdown.
   a) **Scanning**: starts collecting points from a laser scanner.
      i) Min Step Scan To Scan: this value defines the minimum distance between two scan lines.
      ii) Point To Point: this value decides the average point distance on the scan line. It refers to the canter of the scanner's measurement range and changes with increasing or decreasing distance to the object. The value can only be changed in 0.075 mm steps.
   b) **Stationary**: stationary measurement is a mode of measurement where the tracker spends set time (set up by the measurement profile) taking measurements and then returns a point.
      i) Measurement profile: Fast, Standard, Precise according to the following table:

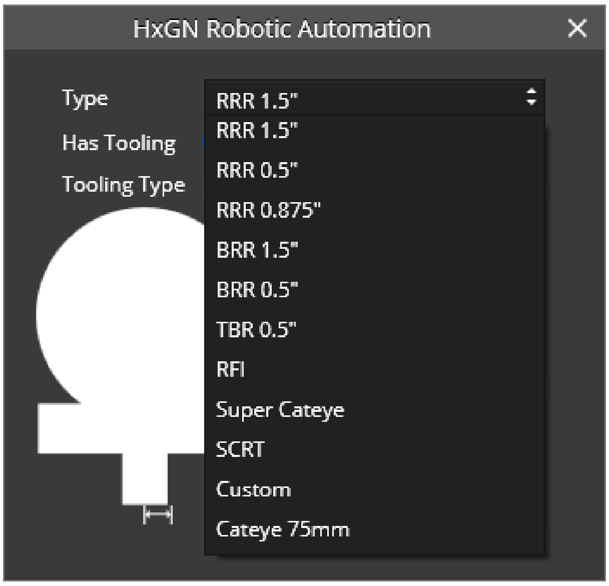
Profile	Measurement Time	User for
Fast	500 ms	Controlled environment
Standard	2000 ms – 4000 ms (distance dependent)	AT960 achieves specified accuracy in a controlled environment
Precise	10000 ms	Highest accuracy – Sensor checks – Compensations

   c) **Stable**: stable measurement will trigger measurement shot when conditions on stable time and space are match.
      i) Stable time: change the time probe must be stable to take a measurement shot.
      ii) Stable Space: Change the space probe must be stable to take a measurement shot.
   d) **Continuous Time**: continuous time mode takes measurement shots based on the frequency given on continuous frequency.
   e) **Continuous Distance**: takes measurement shots based on the distance given on continuous frequency.

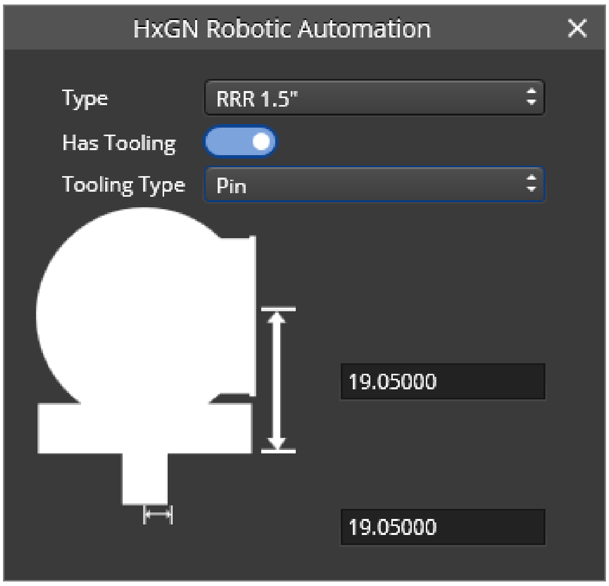


9) **Tool Settings**: shows the tool attached and its options (if any).

Tool settings are only available if tool attached is a reflector (and the Tracker is locked on that reflector). The user can choose the type of reflector from the list or create a custom one and define if there is a tooling and the tooling type (Flat, Edge, Pin) and enter the correct values of the reflector.


HxGN Robotic Automation

Type: RRR 1.5"


HxGN Robotic Automation

Type: RRR 1.5"

Has Tooling: on

Tooling Type: Pin

19.05000

19.05000

10) Buttons to manage smart features (see chapter 7.1.1): **go to previous**, **reset current**, **go to next**.

8.10.3 Moving a cell item dragging it in the 3D environment

To move the machines using the 3D environment you can click and drag the different joints (either in general or using the drag wheel that appears). You can also move the tool itself via the same method or by using the gizmo that appear when choosing it. By moving the tool, the rest of the joint will adjust automatically according to the machine kinematics to reach the desired position.

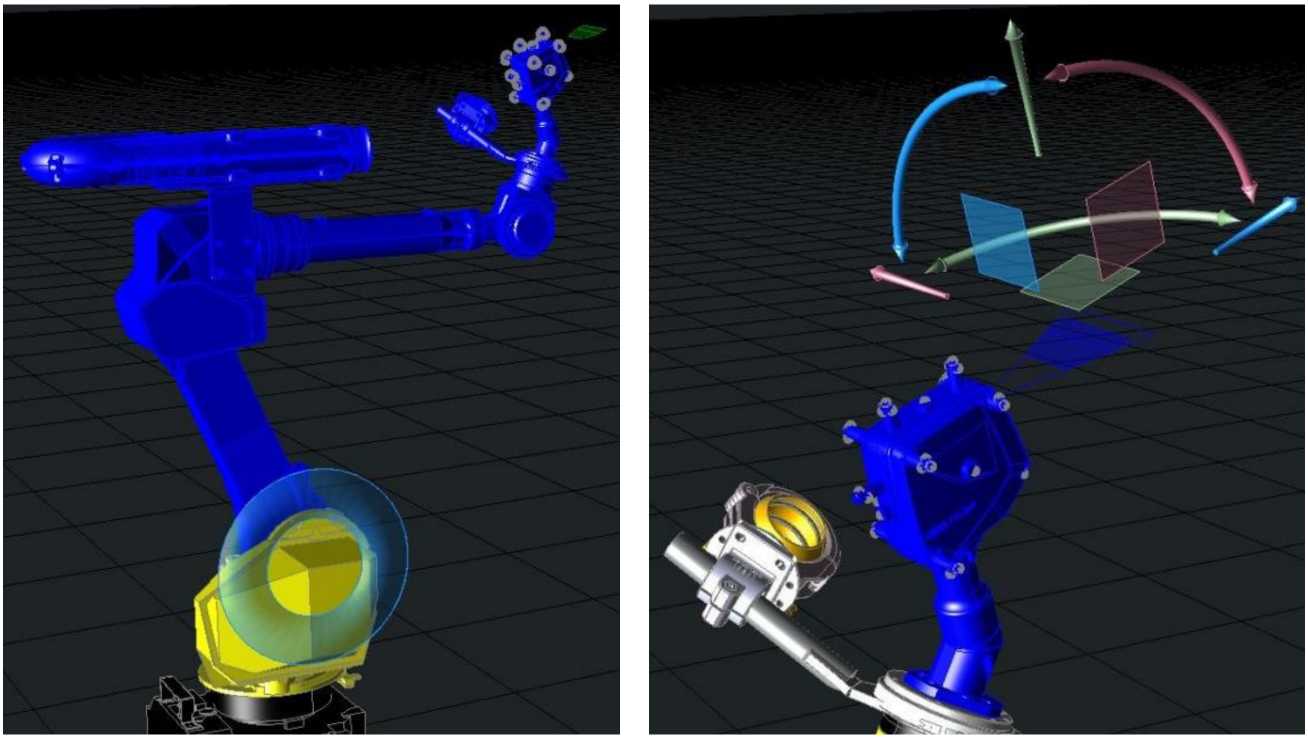
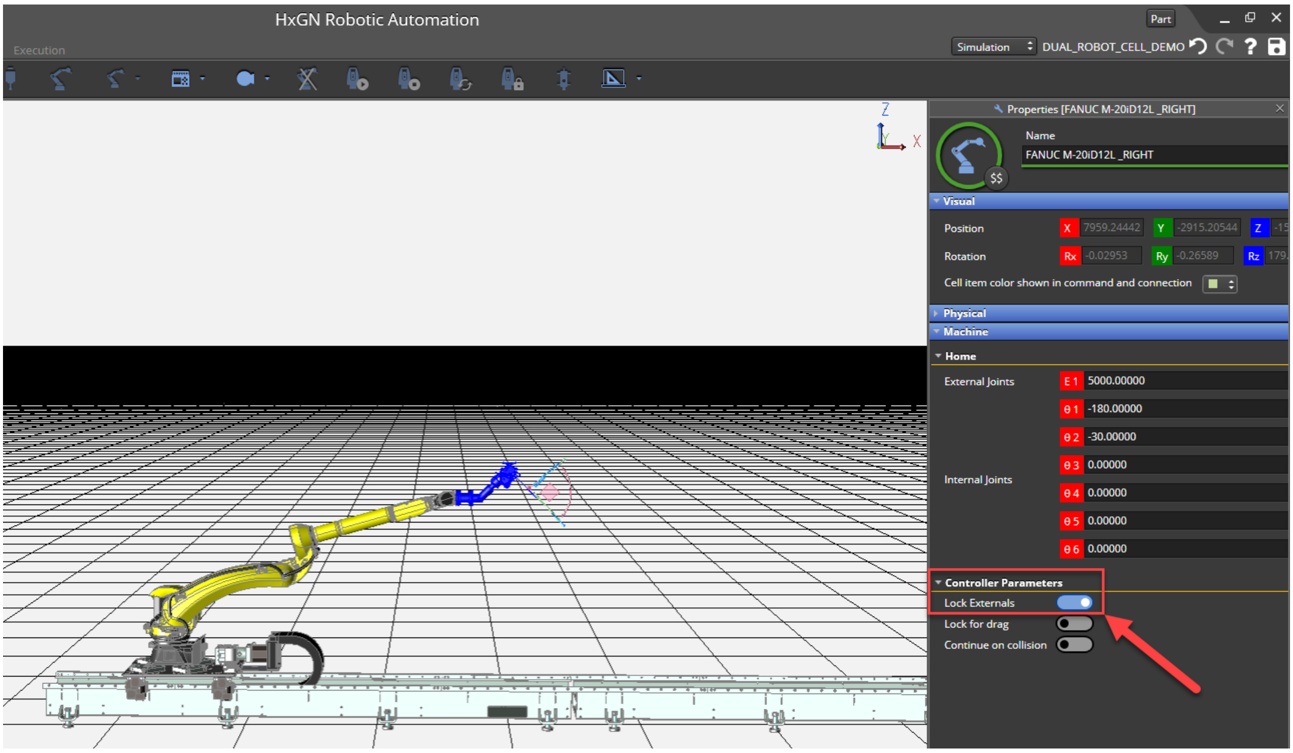


Figure 8-5*: Example of the 3D drag wheel on joint 2 to move the robot in the 1D joint space (left) and 3D drag using the tool gizmo to move the robot in the 6D cartesian space (right)*

If a robot is mounted on a slider (see chapter 5.1.3.2 for modelling information), it is possible to 3D drag the tool gizmo considering the overall 7 axes kinematics of the robot including the external axis (slider) or to lock it. Within machine **Properties**, if the *Lock Externals* toggle switch is enabled, when the robot is moved in the scene, the external axis will be locked so it will not move, if it is disabled also the external axis will be moved to bring the robot to the desired position.

In the left picture the robot is stretching to reach the desired position because the external axis is locked while in the right picture the robot can reach the desired position without stretching its joint as the slider can move and support the movement towards the x axis.



HxGN Robotic Automation

Simulation DUAL_ROBOT_CELL_DEMO

Properties [FANUC M-20iD12L _RIGHT]

Name: FANUC M-20iD12L _RIGHT

Visual

Position X 7959.24442 Y -2915.20544 Z -15

Rotation Rx -0.02953 Ry -0.26589 Rz 179.

Cell item color shown in command and connection

Physical

Machine

Home

External Joints E1 5000.00000

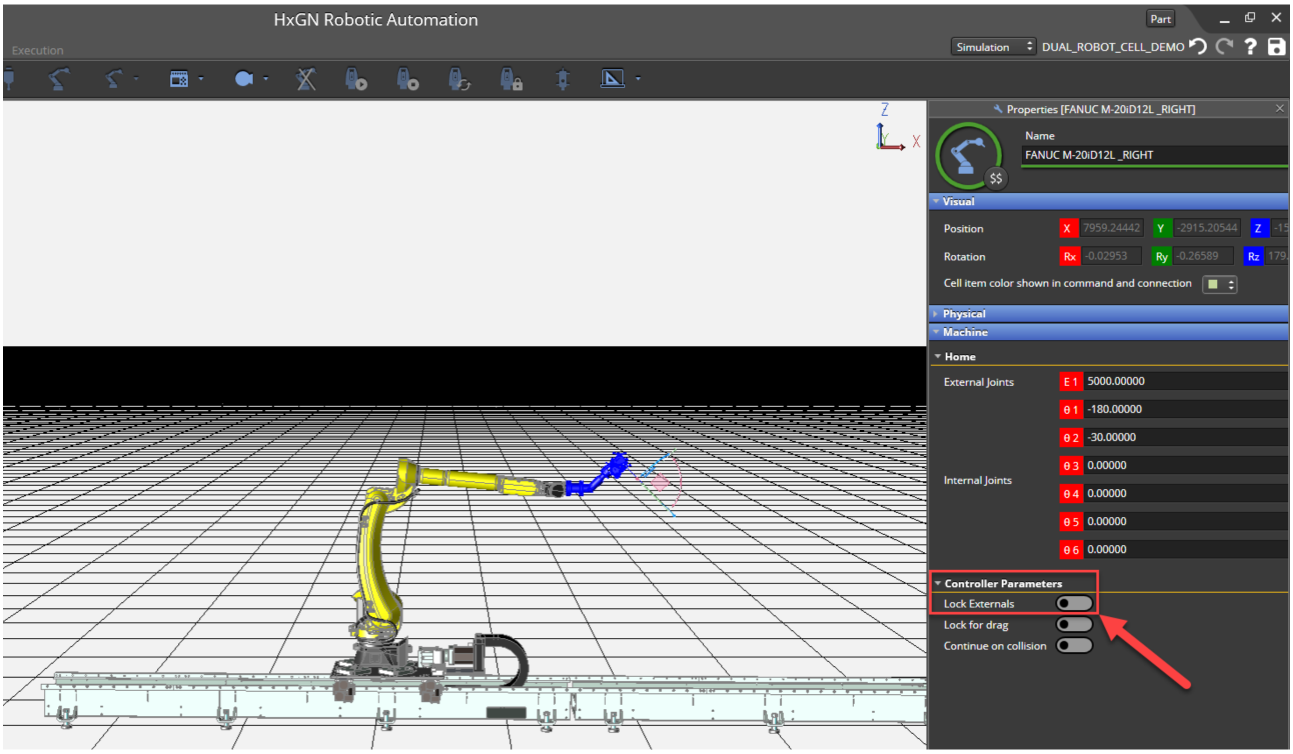
Internal Joints θ1 -180.00000, θ2 -30.00000, θ3 0.00000, θ4 0.00000, θ5 0.00000, θ6 0.00000

Controller Parameters

Lock Externals: on

Lock for drag: off

Continue on collision: off



HxGN Robotic Automation

Simulation DUAL_ROBOT_CELL_DEMO

Properties [FANUC M-20iD12L _RIGHT]

Name: FANUC M-20iD12L _RIGHT

Visual

Position X 7959.24442 Y -2915.20544 Z -15

Rotation Rx -0.02953 Ry -0.26589 Rz 179.

Cell item color shown in command and connection

Physical

Machine

Home

External Joints E1 5000.00000

Internal Joints θ1 -180.00000, θ2 -30.00000, θ3 0.00000, θ4 0.00000, θ5 0.00000, θ6 0.00000

Controller Parameters

Lock Externals: off

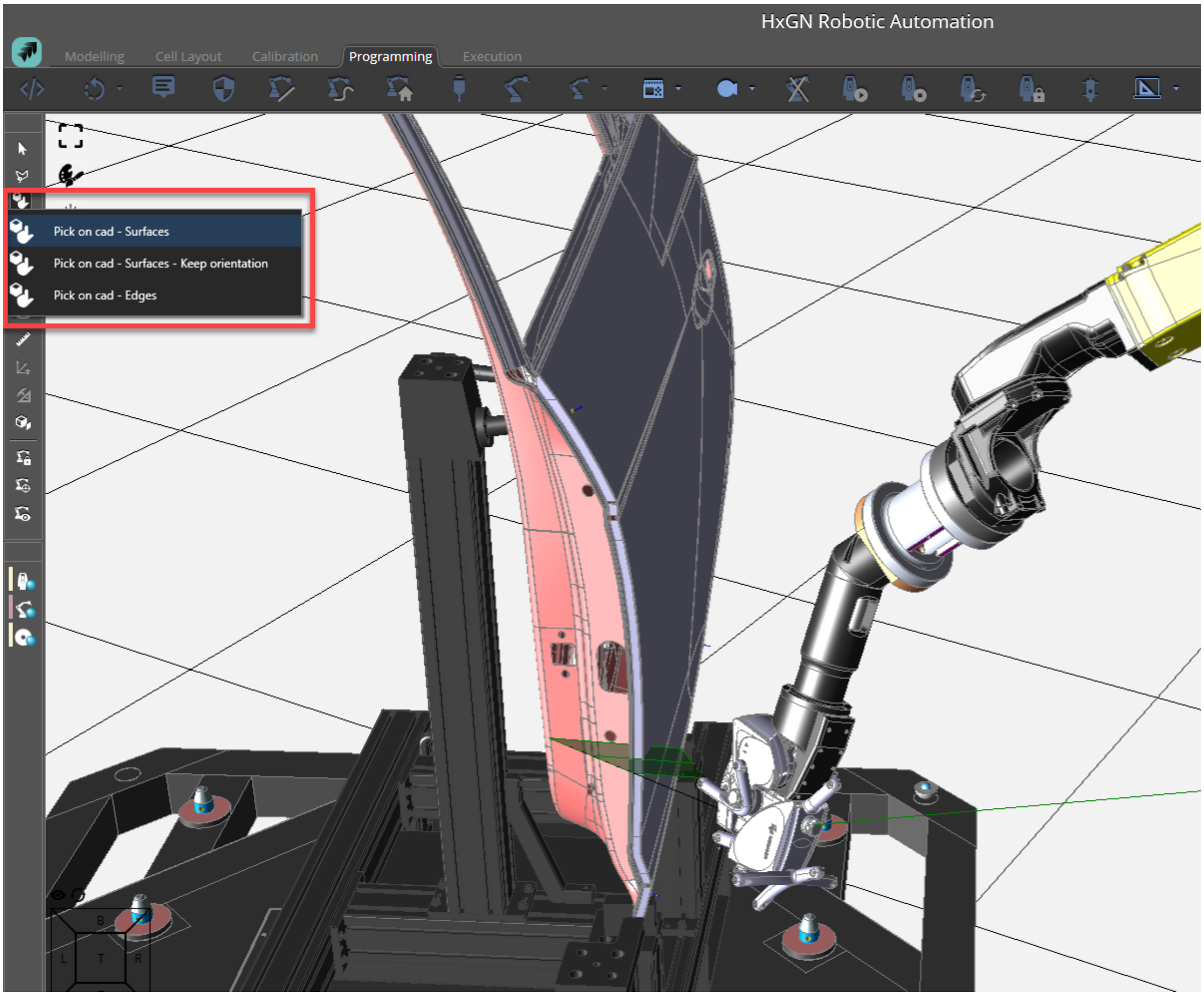
Lock for drag: off

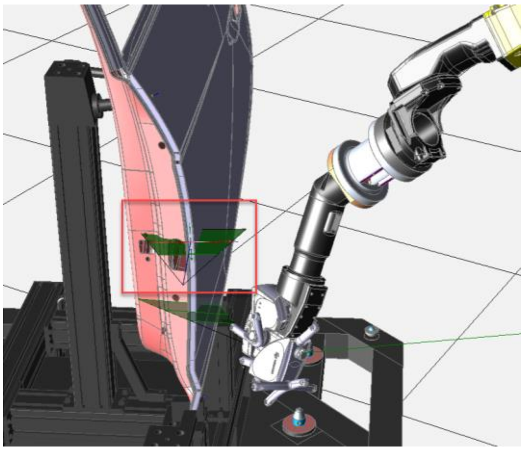
Continue on collision: off

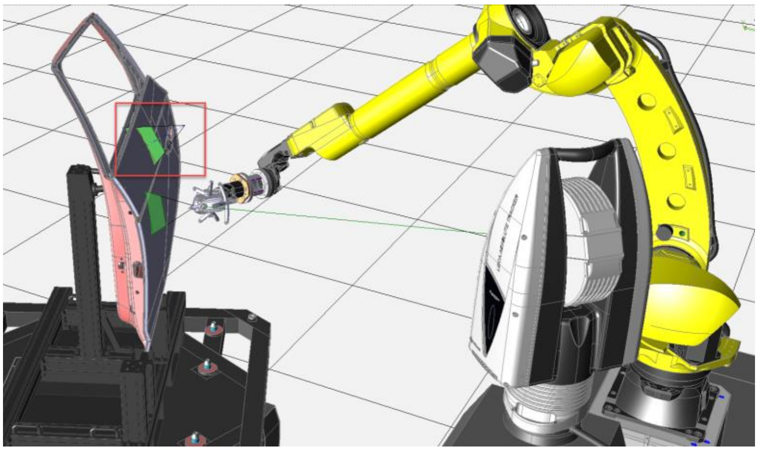
8.10.4 Pick on CAD

Pick on CAD is a special mode that allows the user to select a point either on the surface or on the edge of a CAD model by simply clicking on it.
Pick on CAD offers three ways of picking a point on the CAD model:

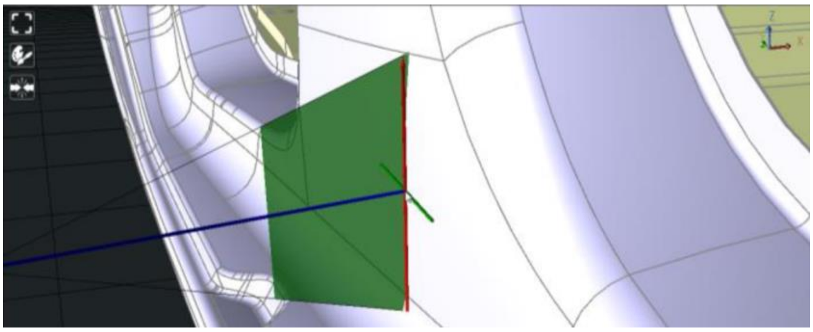
1) **Pick on CAD – Surfaces**: allows to visualize the tool TCP on the CAD surface point and click to move the robot in the specified position.
2) **Pick on CAD – Surfaces – Keep orientation**: same as above but the target orientation is set to the current tool orientation.
3) **Pick on CAD – Edges**: allows to visualize the tool TCP on the CAD surface point and click to move the robot in the specified position.



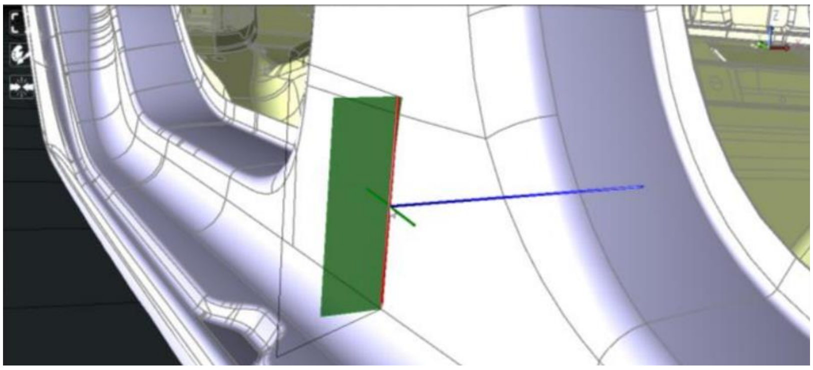




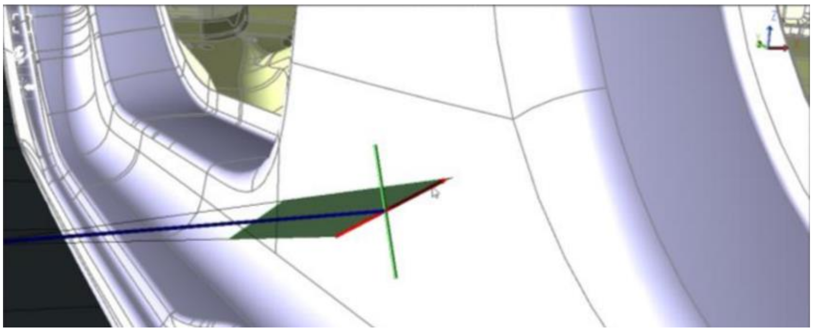
When activated, the scanning range will appear perpendicular to the surface.



To flip the normal [Alt] button needs to be pressed.



To rotate direction [Shift] button needs to be pressed and mouse dragged.

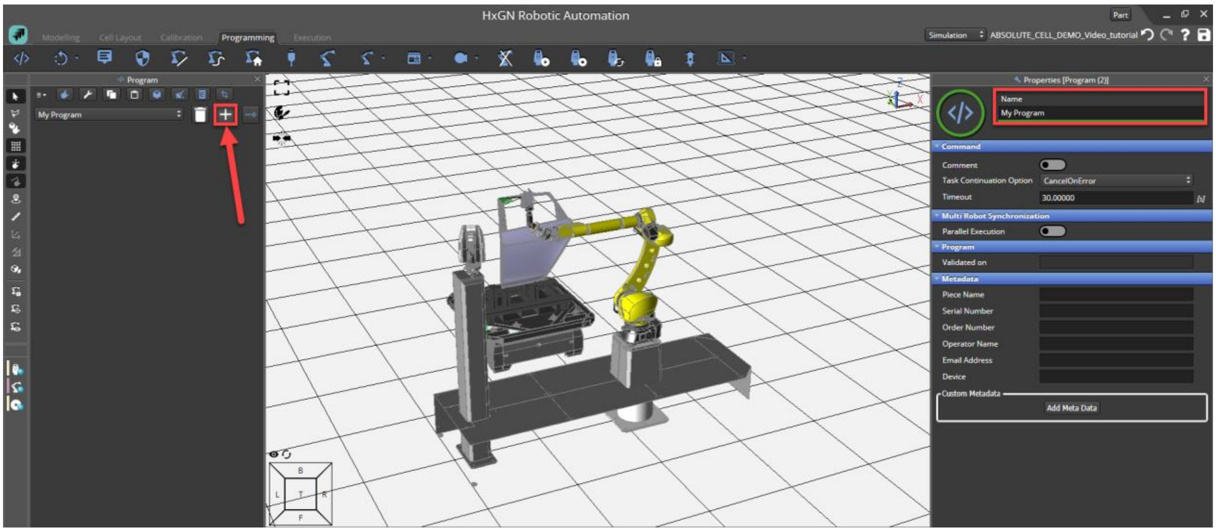


8.10.5 Adding commands

After choosing the position of the robot you can start adding commands from the upper toolbar and begin creating your manual program. When choosing the command notice the right-side window displaying the commands properties in case of any changes you need to make. All available commands are explained in detail in chapter 8.3. In the following example, a basic program is shown where the user creates a simple program to scan a feature by adding manually the necessary commands from the ribbon toolbar.

Example – Single robot cell manual programming

First of all it is needed to click on the plus button and create a new program and call it e.g. “My Program”.



Name: My Program

Command
- Comment: ○
- Task Continuation Option: CancelOnError
- Timeout: 30.00000 [s]

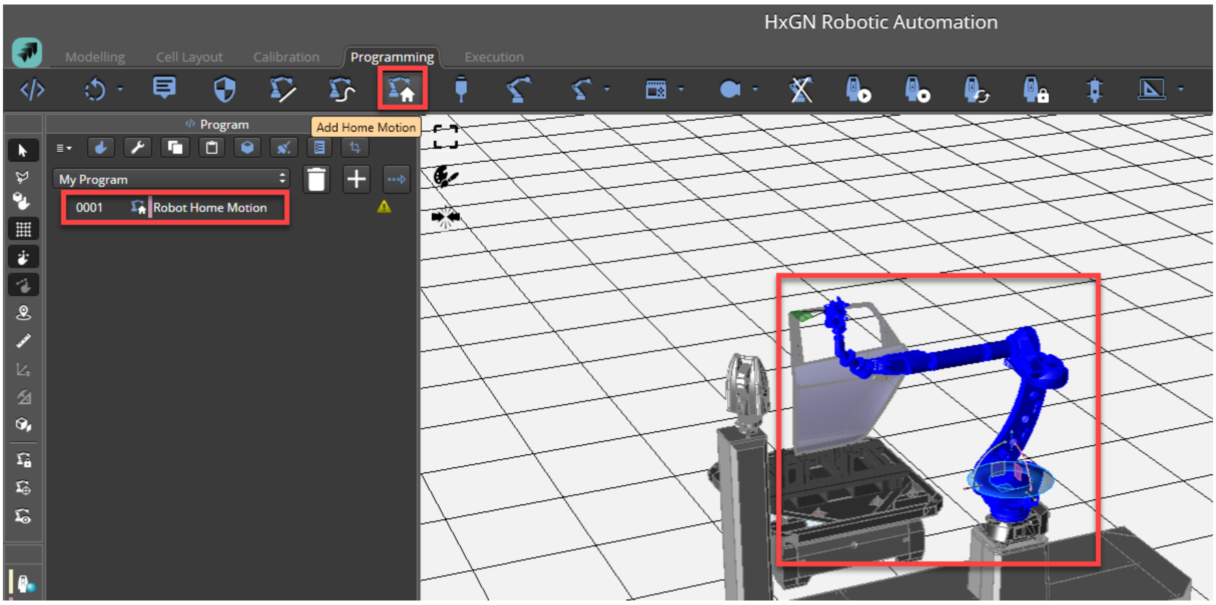
Multi Robot Synchronization
- Parallel Execution: ○

Program
- Validated on:

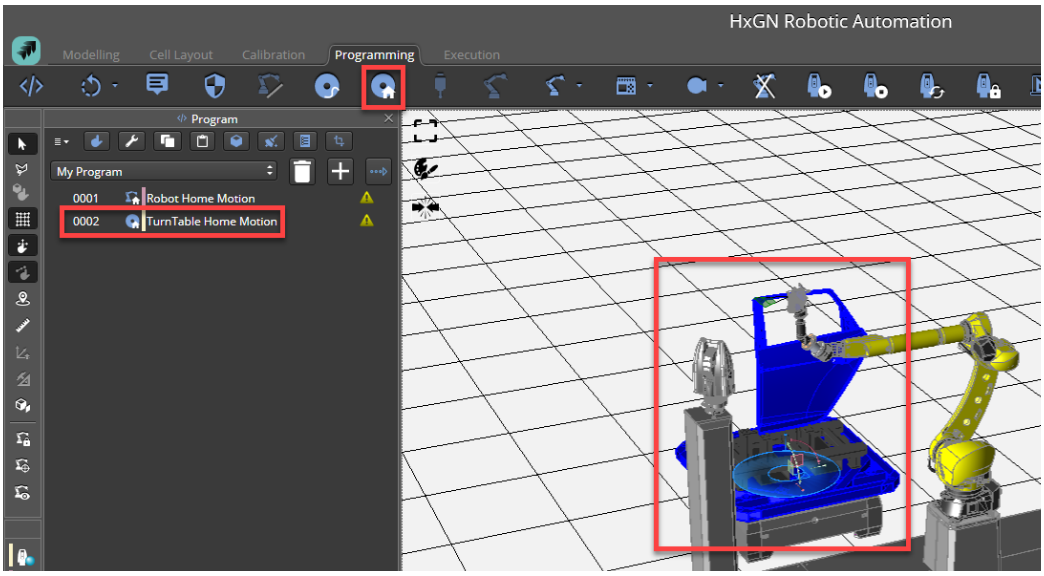
Metadata
- Piece Name:
- Serial Number:
- Order Number:
- Operator Name:
- Email Address:
- Device:

Custom Metadata
- Add Meta Data

Then it is need to add homing motions for the robot and the turn table by selecting them and adding the relative homing commands.



0001 Robot Home Motion

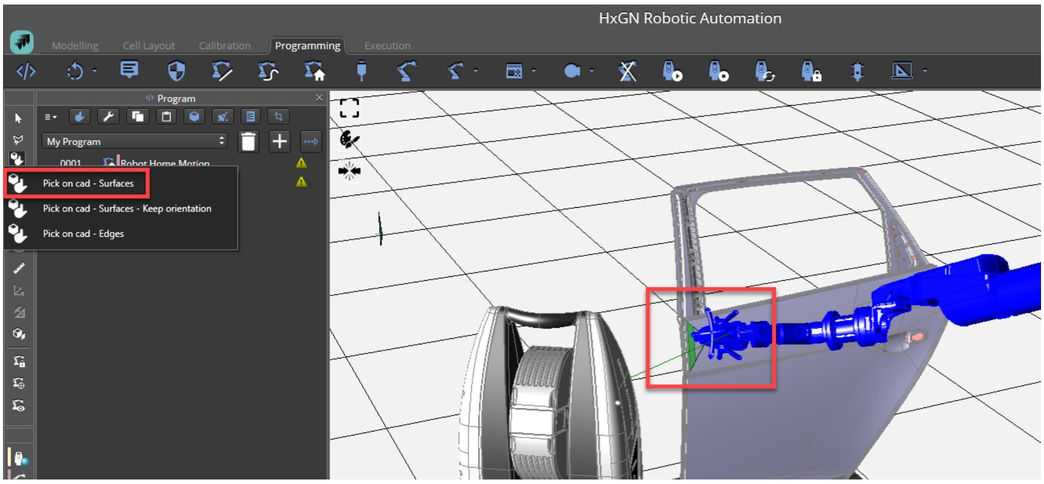


My Program

0001 Robot Home Motion

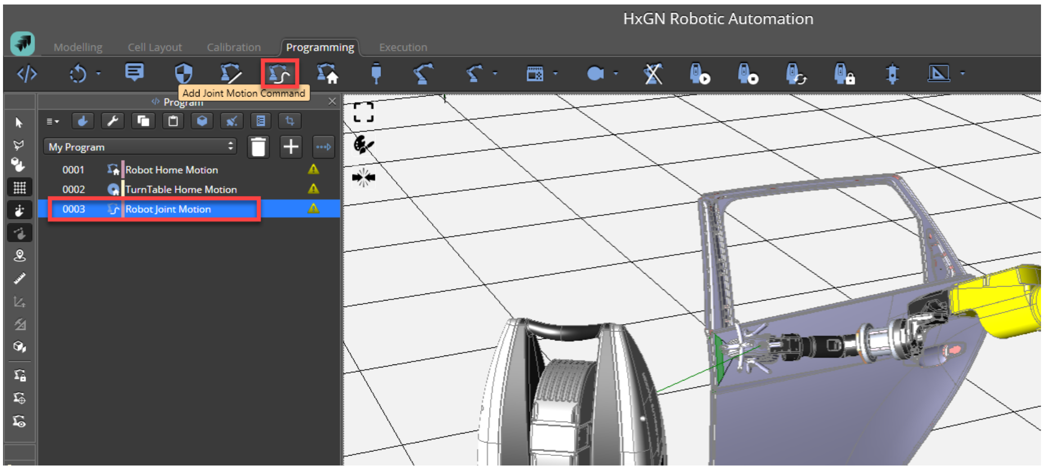
0002 TurnTable Home Motion

At this point, it is possible to move the robot e.g. using the Pick on CAD functionality to a specific point on the CAD surface where the scan will start and then add a joint motion command to take the robot to that position.



My Program

0001 Robot Home Motion



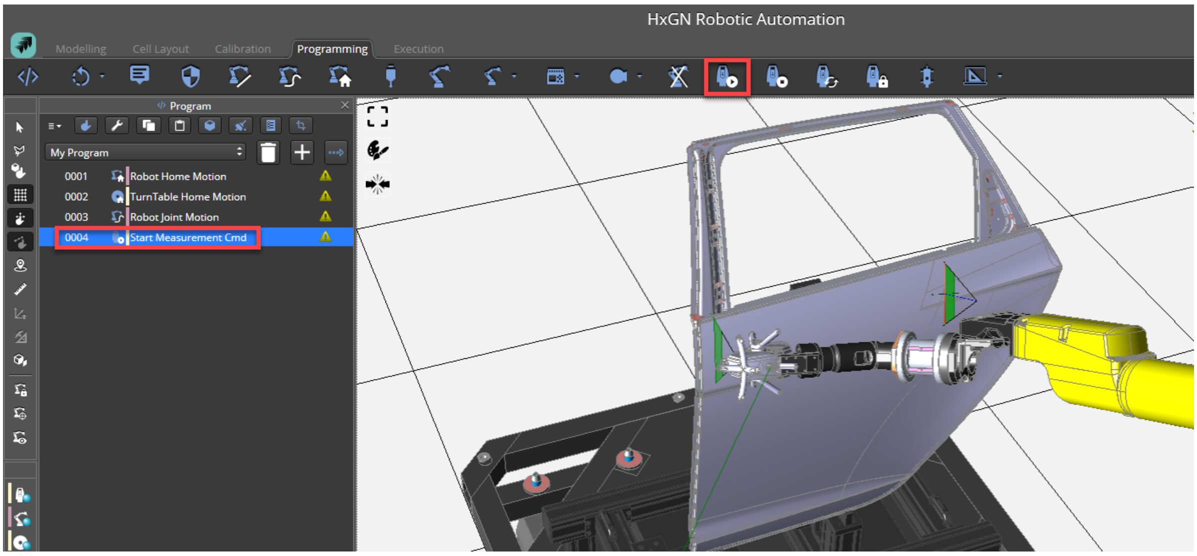
My Program

0001 Robot Home Motion

0002 TurnTable Home Motion

0003 Robot Joint Motion

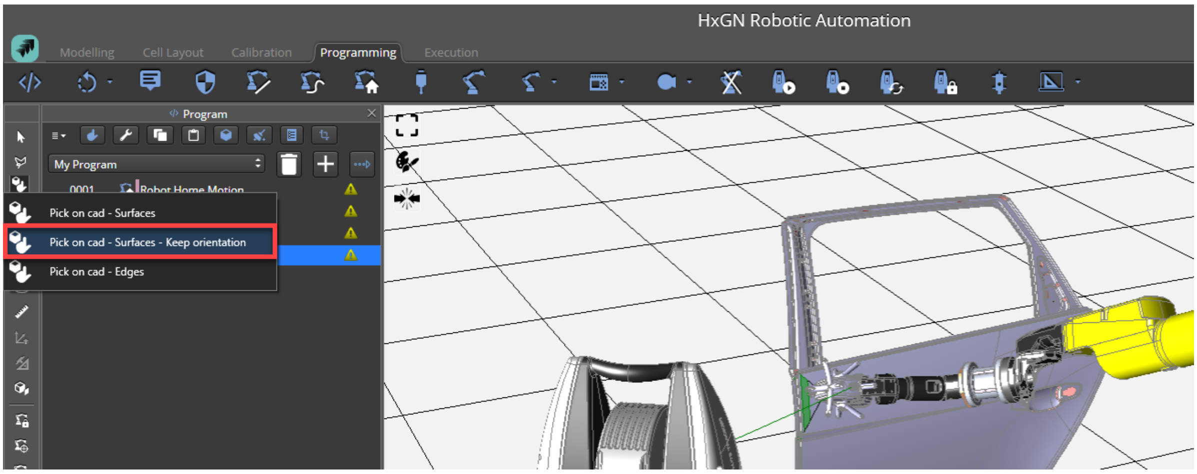
Now it is possible to add a Start Measurement command to start the scanning from this position.

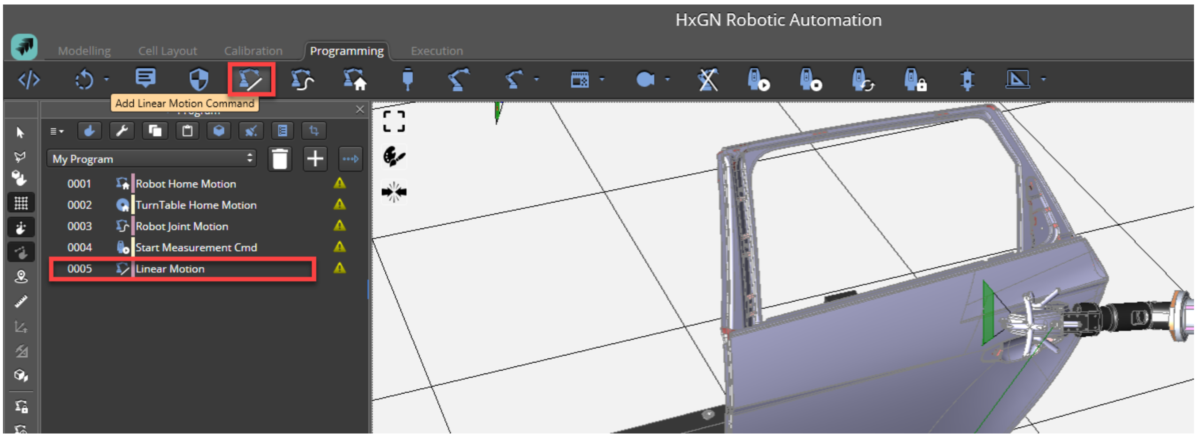


My Program

0001	Robot Home Motion
0002	TurnTable Home Motion
0003	Robot Joint Motion
0004	Start Measurement Cmd

Then using the Pick on CAD – Keep orientation functionality is then possible to specify the next position that the robot will reach using a linear movement while scanning. After clicking on the next position on the CAD, a linear motion command to can be added to the program.

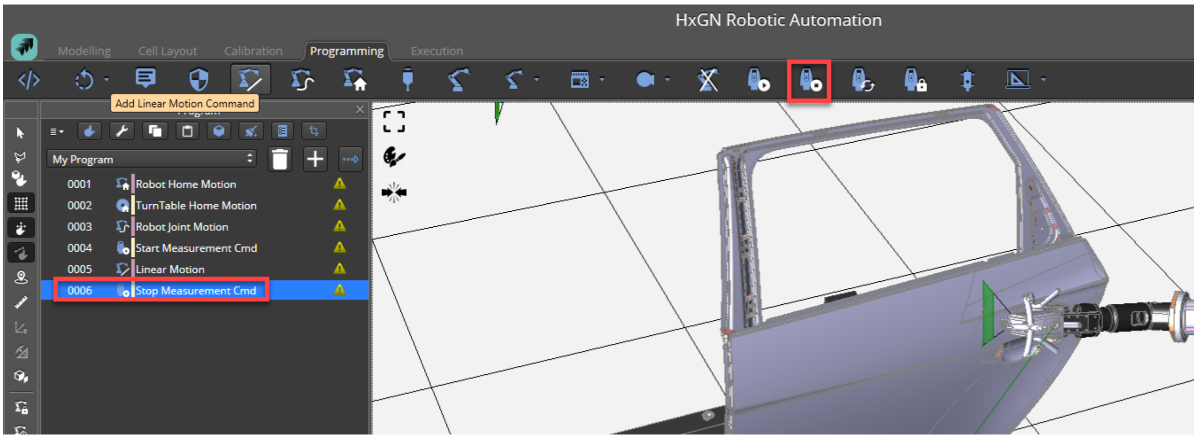




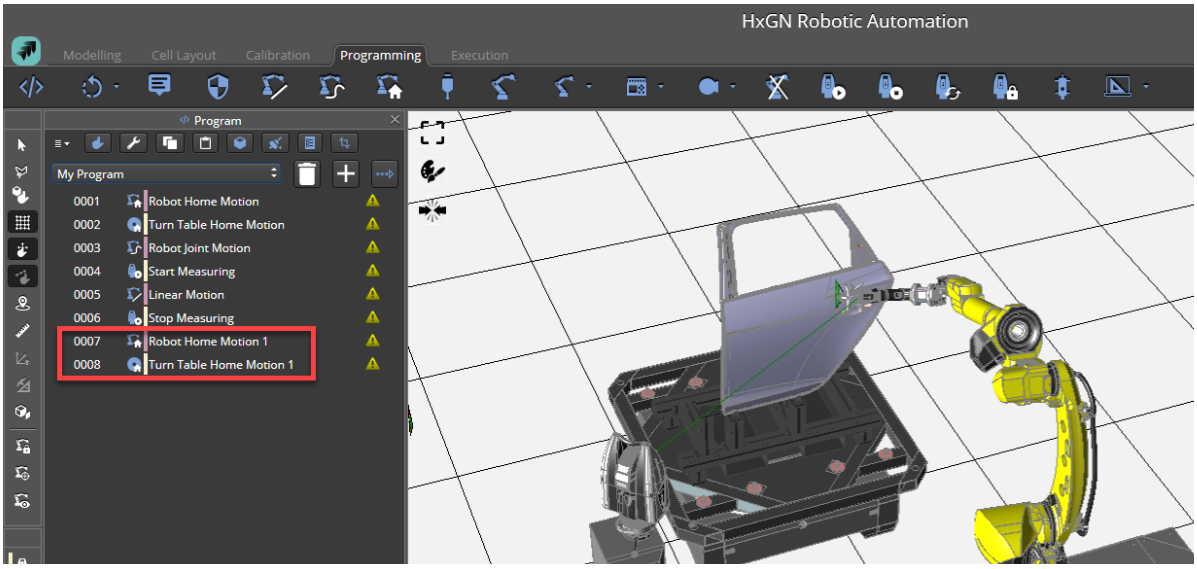
My Program

0001	Robot Home Motion
0002	TurnTable Home Motion
0003	Robot Joint Motion
0004	Start Measurement Cmd
0005	Linear Motion

Finally a Stop Measurement command can be added to stop the scanning, as well as homing motions to take the robot and the turn table to their initial positions.



0001 Robot Home Motion
0002 TurnTable Home Motion
0003 Robot Joint Motion
0004 Start Measurement Cmd
0005 Linear Motion
0006 Stop Measurement Cmd

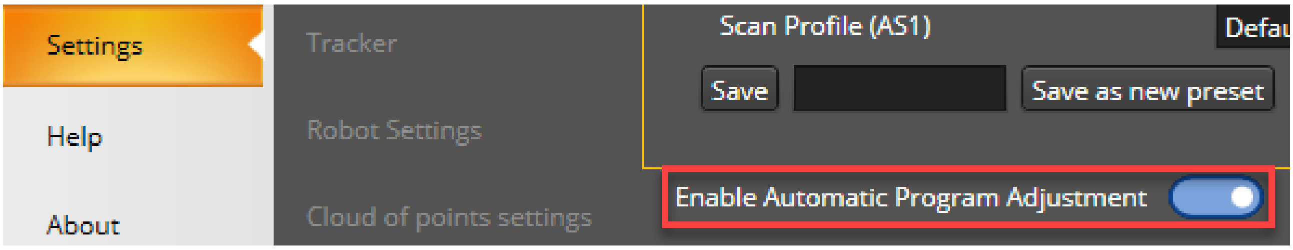


0001 Robot Home Motion
0002 Turn Table Home Motion
0003 Robot Joint Motion
0004 Start Measuring
0005 Linear Motion
0006 Stop Measuring
0007 Robot Home Motion 1
0008 Turn Table Home Motion 1

8.11 Automatic Program Adjustment

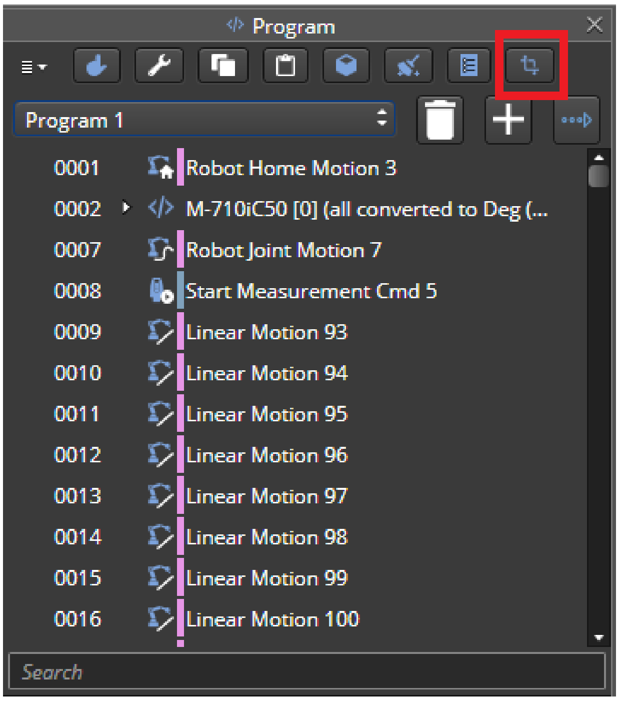
This feature re-calculates an existing program when there is a change in the model. This allows the user to migrate old programs according to a new model. It may happen that a program cannot be fully migrated due to the new model (e.g.: collisions, parts out of reach, etc.). In that case, the new program indicates which motions could not be migrated properly and has to be resolved manually by the user.

The option can be activated under Settings | Planning with the relative toggle switch. By default it is disabled.

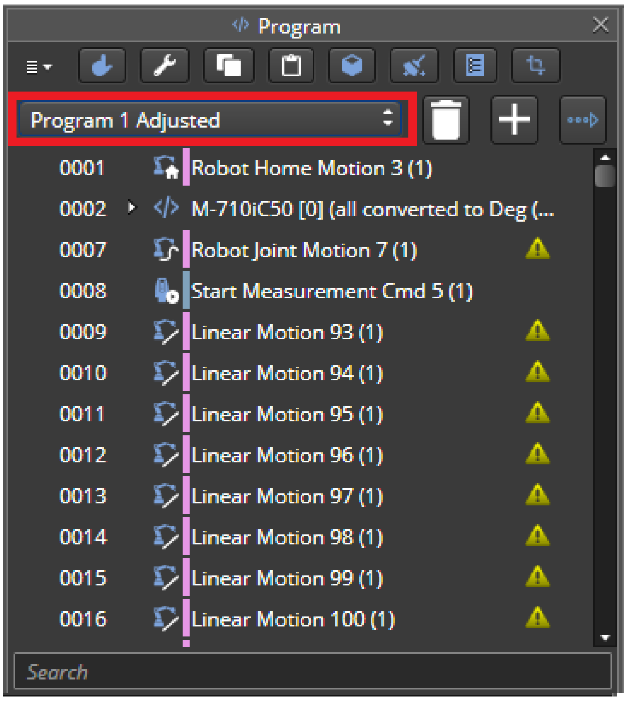


Enable Automatic Program Adjustment

In the Programming tab, you can find the Automatic Program Adjustment button. After selecting an existing program, if you click the button a new program is created. If there was a modification in the model, the new program will adjust the previous one in order to scan the same area if it is possible. Otherwise, the resultant program is the same.



```
Program
Program 1
0001  Robot Home Motion 3
0002  M-710iC50 [0] (all converted to Deg (...
0007  Robot Joint Motion 7
0008  Start Measurement Cmd 5
0009  Linear Motion 93
0010  Linear Motion 94
0011  Linear Motion 95
0012  Linear Motion 96
0013  Linear Motion 97
0014  Linear Motion 98
0015  Linear Motion 99
0016  Linear Motion 100
Search
```

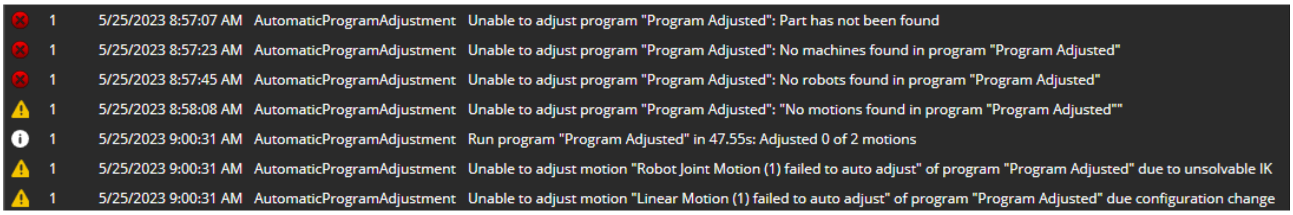


```
Program
Program 1 Adjusted
0001  Robot Home Motion 3 (1)
0002  M-710iC50 [0] (all converted to Deg (...
0007  Robot Joint Motion 7 (1)         ⚠
0008  Start Measurement Cmd 5 (1)
0009  Linear Motion 93 (1)             ⚠
0010  Linear Motion 94 (1)             ⚠
0011  Linear Motion 95 (1)             ⚠
0012  Linear Motion 96 (1)             ⚠
0013  Linear Motion 97 (1)             ⚠
0014  Linear Motion 98 (1)             ⚠
0015  Linear Motion 99 (1)             ⚠
0016  Linear Motion 100 (1)            ⚠
Search
```

The execution time of adjusting automatically a program should be within the order of a few seconds at most for long projects. For simple robots even less than a second.

Warning: it is always recommended validating the newly adjusted program to avoid potential collisions.

All information about the automatic adjustment like errors, warnings, and info messages are shown to the user in the Status window as per example below.

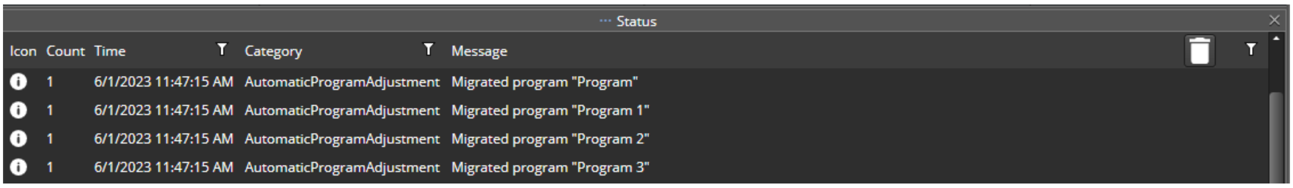


```
⊗ 1  5/25/2023 8:57:07 AM  AutomaticProgramAdjustment  Unable to adjust program "Program Adjusted": Part has not been found
⊗ 1  5/25/2023 8:57:23 AM  AutomaticProgramAdjustment  Unable to adjust program "Program Adjusted": No machines found in program "Program Adjusted"
⊗ 1  5/25/2023 8:57:45 AM  AutomaticProgramAdjustment  Unable to adjust program "Program Adjusted": No robots found in program "Program Adjusted"
⚠ 1  5/25/2023 8:58:08 AM  AutomaticProgramAdjustment  Unable to adjust program "Program Adjusted": "No motions found in program "Program Adjusted""
ⓘ 1  5/25/2023 9:00:31 AM  AutomaticProgramAdjustment  Run program "Program Adjusted" in 47.55s: Adjusted 0 of 2 motions
⚠ 1  5/25/2023 9:00:31 AM  AutomaticProgramAdjustment  Unable to adjust motion "Robot Joint Motion (1) failed to auto adjust" of program "Program Adjusted" due to unsolvable IK
⚠ 1  5/25/2023 9:00:31 AM  AutomaticProgramAdjustment  Unable to adjust motion "Linear Motion (1) failed to auto adjust" of program "Program Adjusted" due configuration change
```

8.11.1 Migration of old Programs

Since the old programs do not have the current parameters, we create those parameters when loading the project. That allows us to adjust the program automatically if it is needed. Recall that to preserve those changes it is mandatory to save the project before closing RA.

Every time an old program is migrated, the following message is shown to the user in the Status window:



Icon	Count	Time	Category	Message
ⓘ	1	6/1/2023 11:47:15 AM	AutomaticProgramAdjustment	Migrated program "Program"
ⓘ	1	6/1/2023 11:47:15 AM	AutomaticProgramAdjustment	Migrated program "Program 1"
ⓘ	1	6/1/2023 11:47:15 AM	AutomaticProgramAdjustment	Migrated program "Program 2"
ⓘ	1	6/1/2023 11:47:15 AM	AutomaticProgramAdjustment	Migrated program "Program 3"

8.11.2 Additional notes

1) If a part is located on another machine, the original program will not be adjusted to the new machines, since exclusively the machines of the original program will be considered. The programs contain information on the machines by their ID, which is never changed when the program is adjusted.

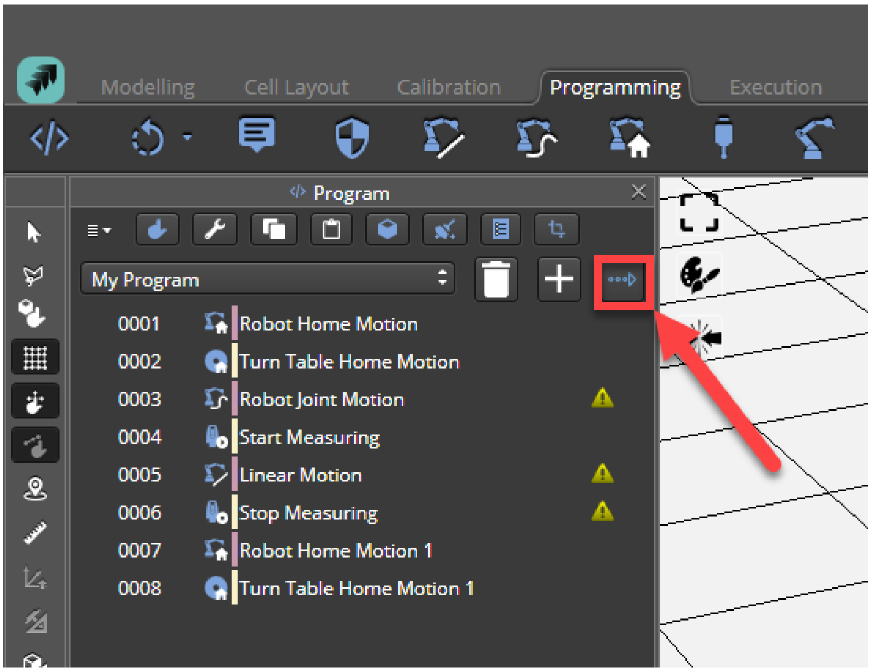
2) If a motion cannot be adjusted, check out that it is reachable by doing a Pick On Cad (see chapter 8.10.4 for further explanation of this functionality).

3) For an adjusted program, the adjusted motions may lead to collisions. For linear motions, collisions can occur between the internal structure of the robot and the environment; for joint motions, collisions can occur also with the tool. This is due to the fact that the new program is not checked for collisions, this is for the user to fix with a manual program. Consider manually changing the configuration of the robot at the motions with collisions to find valid adjusted motions.

4) If a program is generated with no motions successfully adjusted, the adjusted program will be exactly the same as the original.

8.12 Validation

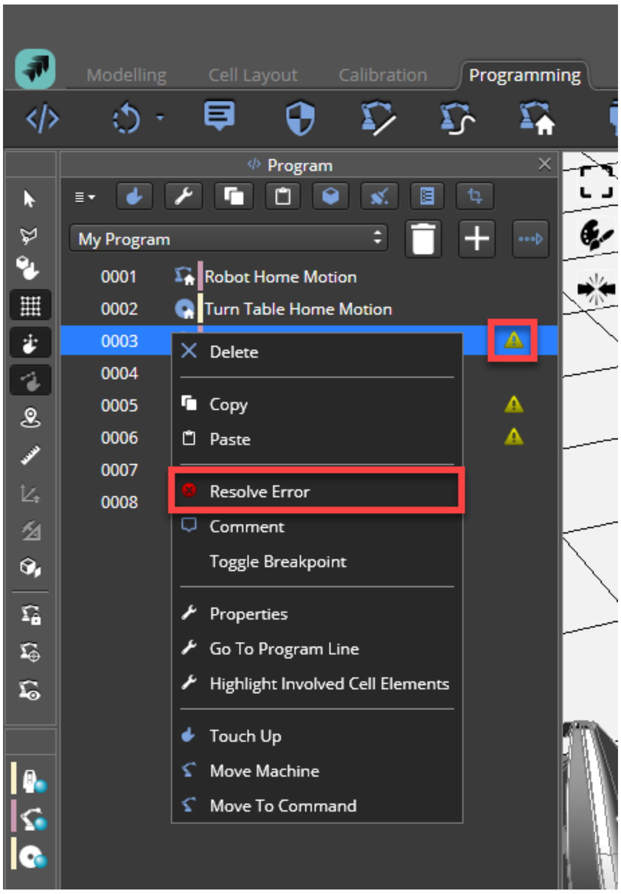
After a program is generated, it is needed to validate it by clicking on the blue arrow above the program.



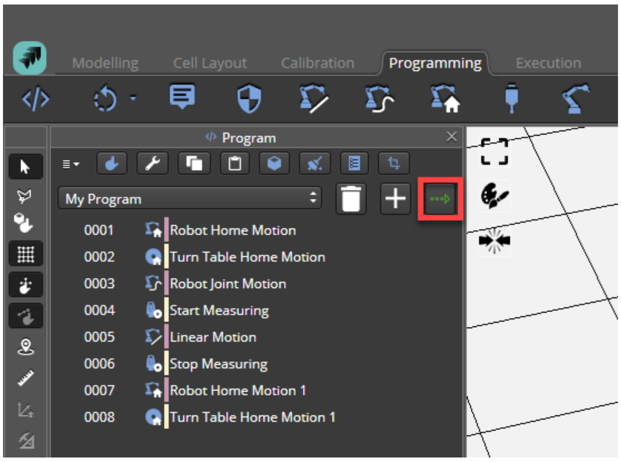
0001 Robot Home Motion
0002 Turn Table Home Motion
0003 Robot Joint Motion
0004 Start Measuring
0005 Linear Motion
0006 Stop Measuring
0007 Robot Home Motion 1
0008 Turn Table Home Motion 1

The validation checks for collisions and other issues in the program (e.g. a reflector is obstructed and cannot be measured by the tracker) and will let you know by displaying a warning sign next to the problematic command in case it finds any.

In case of a collision, you can either fix it manually or use the option **Resolve Error** within the properties shown after right-clicking on the problematic program line.



Once the Validation passes the arrow will turn green showing that the program is safe to run.



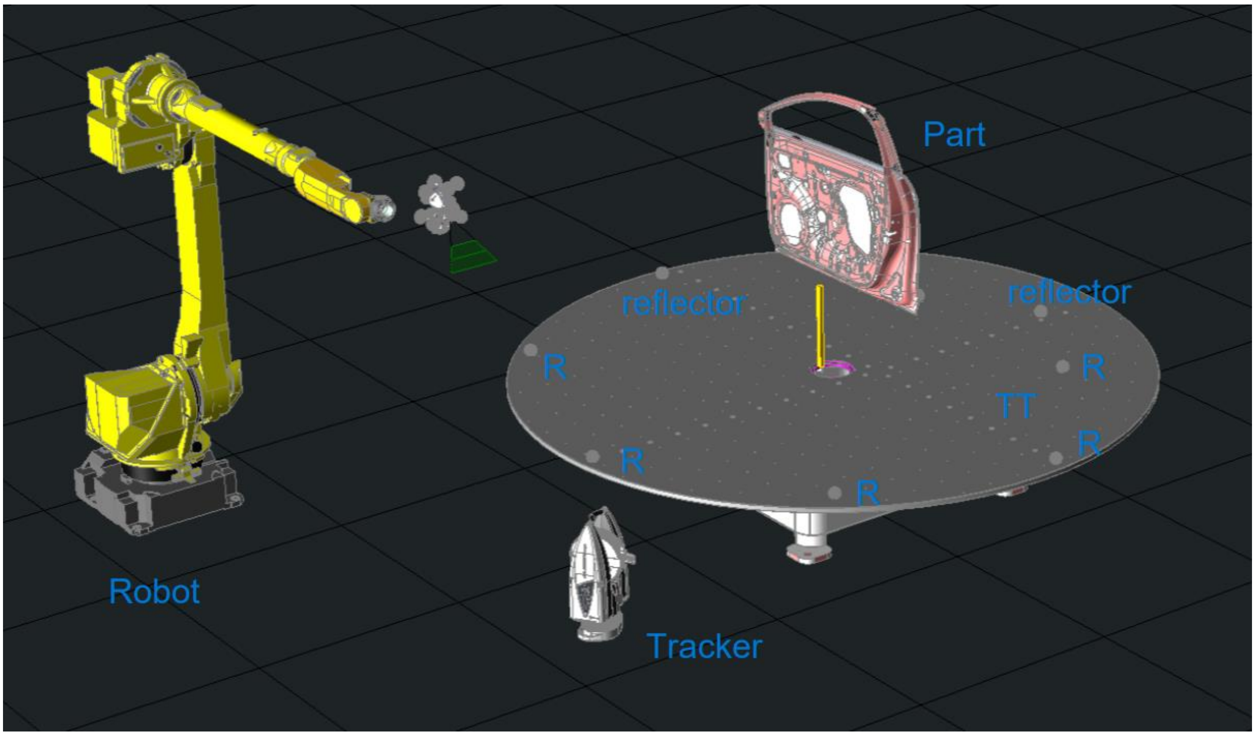
8.13 Notable programming use cases

In this section five notable scenarios are described together with the best practices to follow to create programs to cover such situations.

8.13.1 Single robot cell with turn table and stationary tracker

Considering a cell with the following configuration:

- 1 stationary tracker
- 1 robot
- 1 turn table
- 1 group of movable reflectors (attached to turn table)

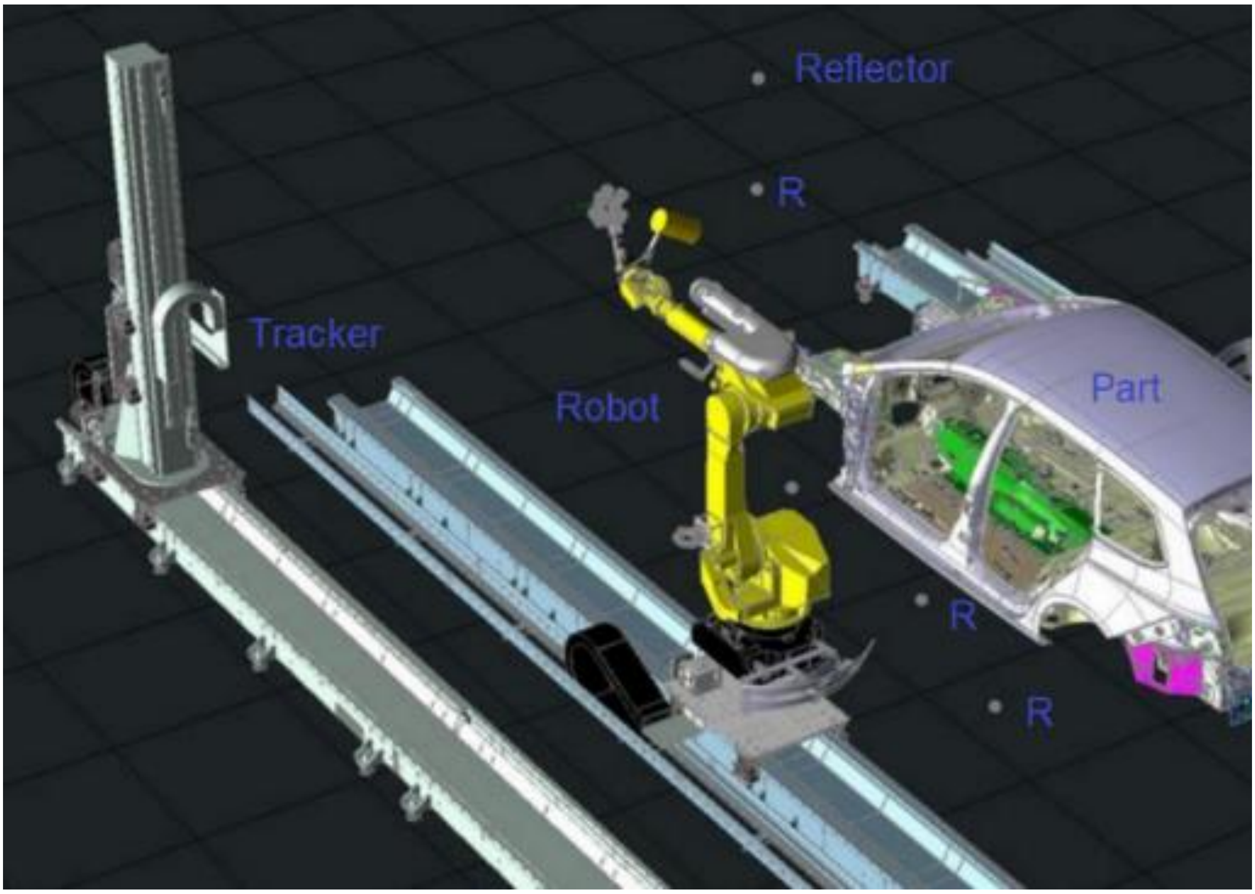


The user can create a program manually adding the necessary commands or using the path planning both in automatic or semi-automatic mode.
In all cases, the program needs to have the structure depicted in **Figure 8-6** resembling the process that the automatic path planning is computing and according to the rules for compensation defined in chapter 8.3.6.
In automatic or semi-automatic mode, it is necessary to follow the steps defined in the relative chapters.

8.13.2 Single robot cell with stationary part holder and movable tracker

Considering a cell with the following configuration:

- 1 movable tracker over a slider
- 1 robot
- 1 stationary part holder
- 1 group of fixed reflectors



The same considerations above apply to this use case as the movable slider also inserts a change in the relationship between tracker and part, so the same rules apply for the compensation part as described in **Figure 8-6**.

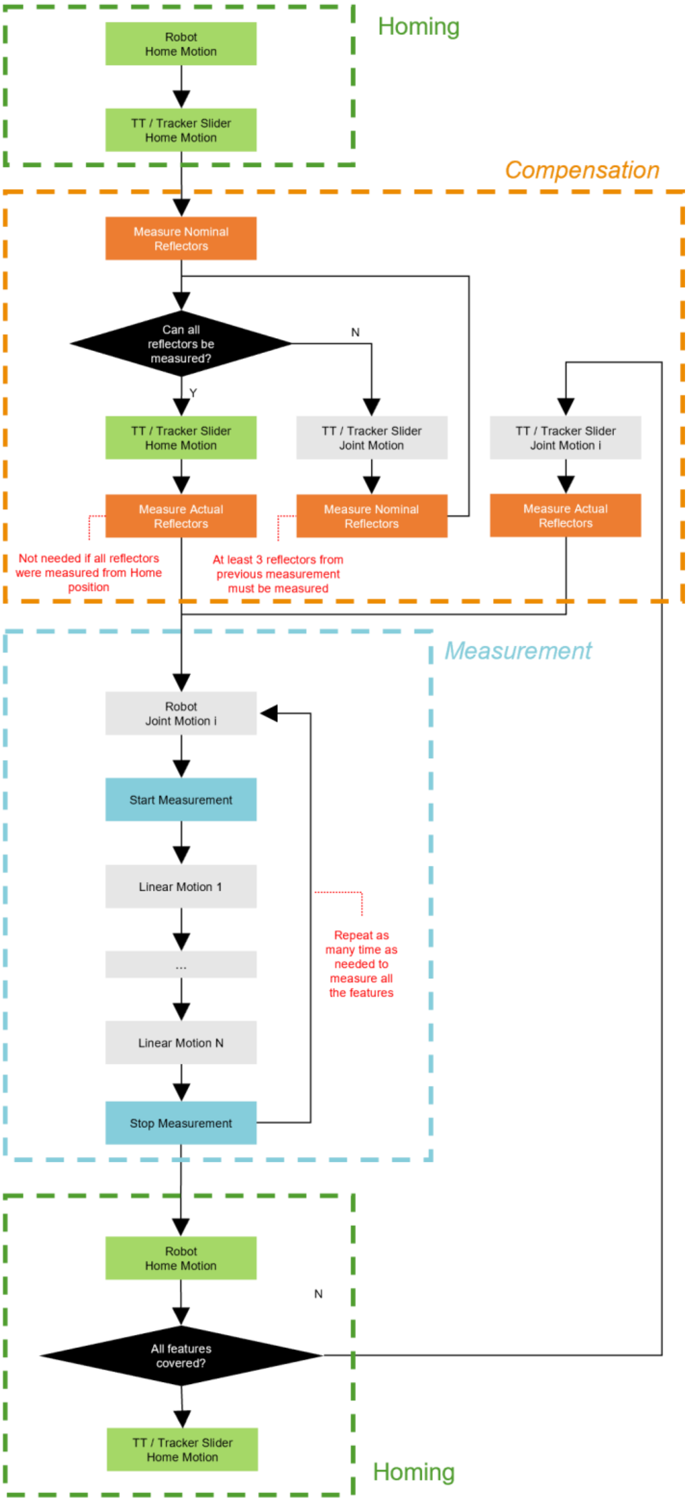
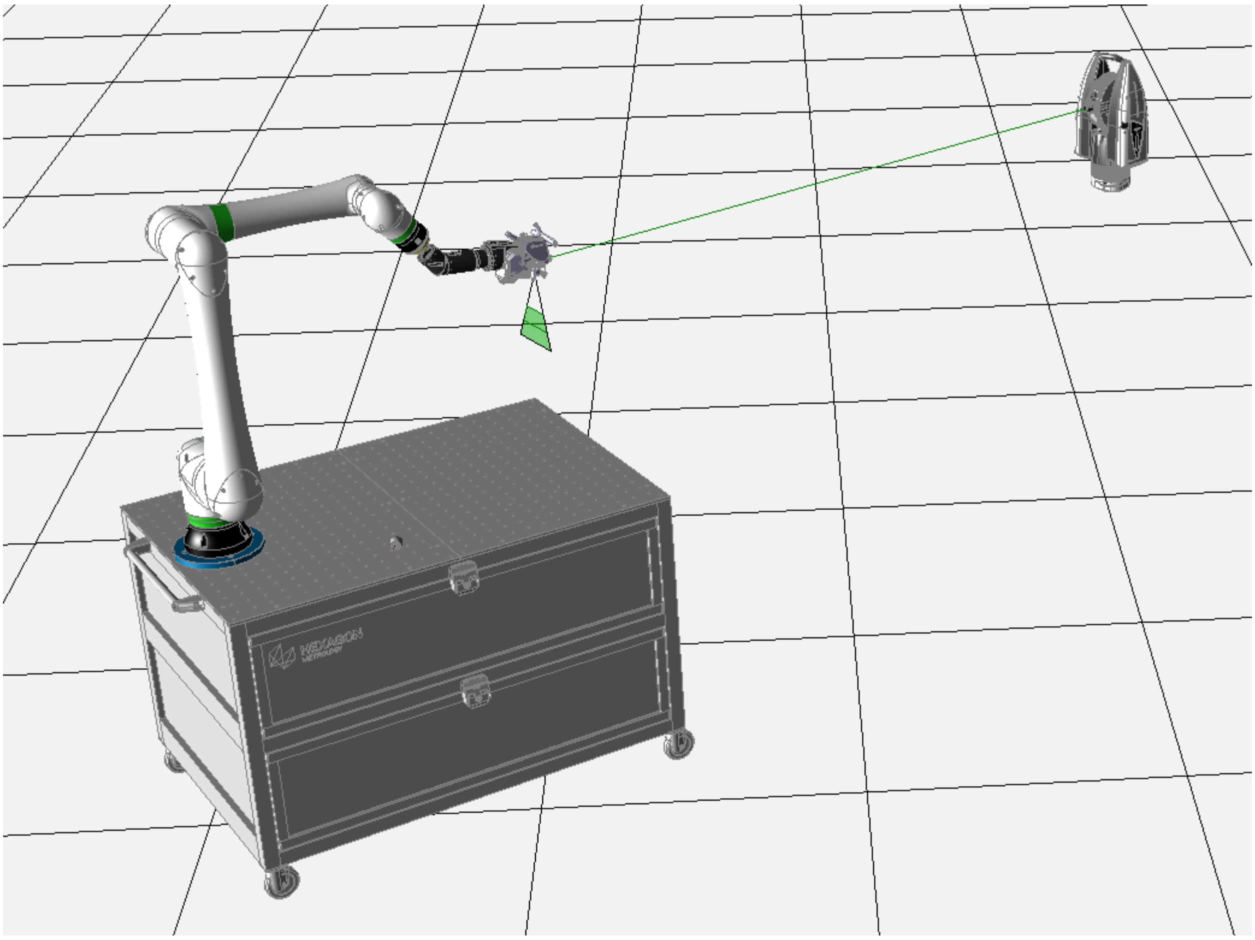

Homing
Robot Home Motion
TT / Tracker Slider Home Motion
Compensation
Measure Nominal Reflectors
Can all reflectors be measured?
N
Y
TT / Tracker Slider Home Motion
TT / Tracker Slider Joint Motion
TT / Tracker Slider Joint Motion i
Measure Actual Reflectors
Measure Nominal Reflectors
Measure Actual Reflectors
Not needed if all reflectors were measured from Home position
At least 3 reflectors from previous measurement must be measured
Measurement
Robot Joint Motion i
Start Measurement
Linear Motion 1
...
Linear Motion N
Stop Measurement
Repeat as many time as needed to measure all the features
Robot Home Motion
N
All features covered?
TT / Tracker Slider Home Motion
Homing

***Figure 8-6**: programming workflow for a single robot cell with turn table and stationary tracker (consider "TT" within the relative blocks) and for a single robot cell with stationary part holder and movable tracker (consider "Tracker Slider" within the relative blocks)*

8.13.3 Single robot cell with stationary part holder and stationary tracker

Considering a cell with the following configuration:

- 1 stationary tracker
- 1 robot
- 1 stationary part holder
- 1 group of fixed reflectors

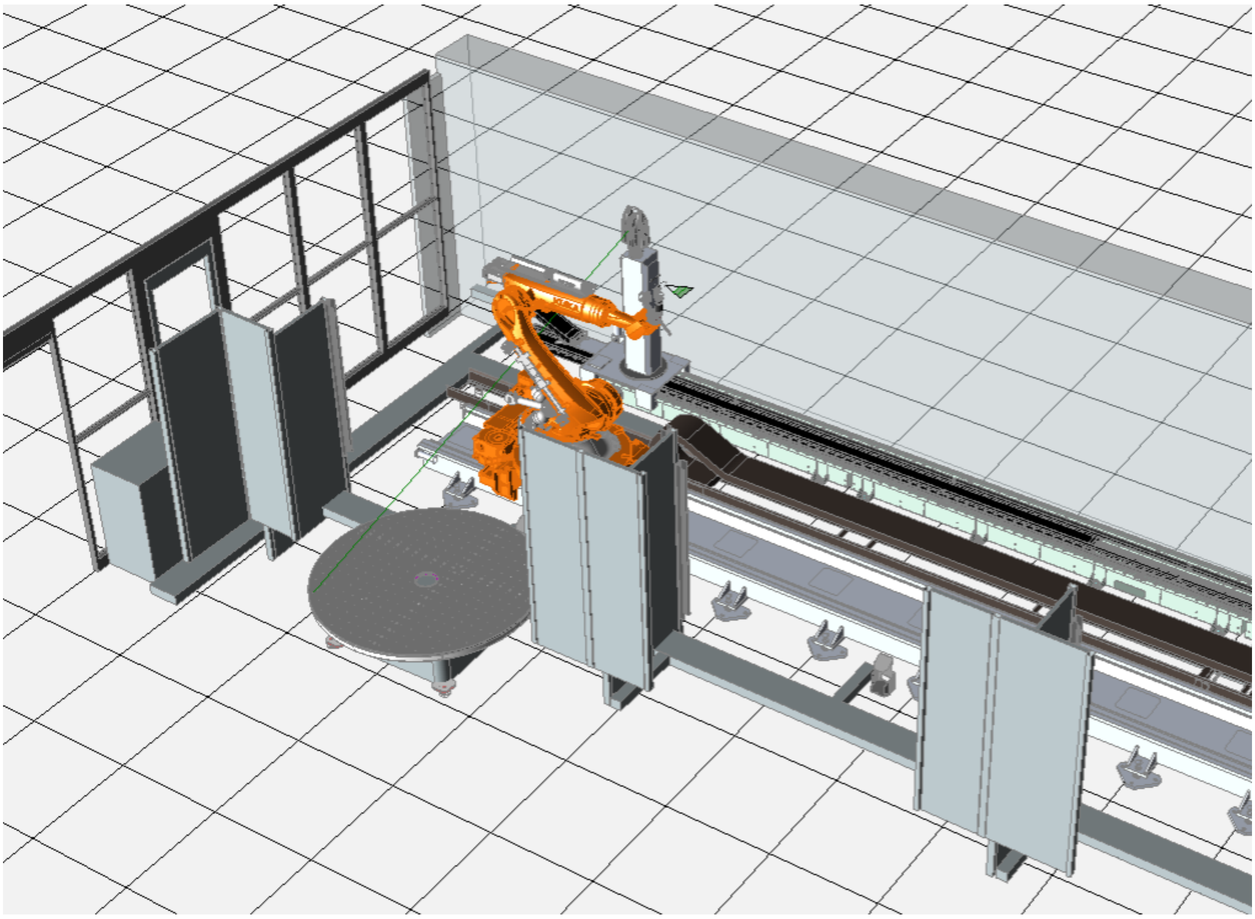


In this case no compensation is necessary as all reflectors must be visible (and therefore measured) from a single tracker position and since the part holder is stationary, there will be no changes in the relationship between tracker and part so the programming workflow is the same as described in the previous sections and depicted in **Figure 8-6** without considering the compensation parts.

8.13.4 Single robot cell with turn table and movable tracker

Considering a cell with the following configuration:

- 1 movable tracker over a slider
- 1 robot
- 1 turn table
- 2 groups of reflectors:
    - movable reflectors (attached to turn table)
    - fixed reflectors



In this case, there are two reasons why there can be a change in the relationship between tracker and part as both the turn table holding the part and the tracker can move.

Since there are two different changes in the relationship between tracker and part, it is needed to have two reflector groups, one for the reflectors mounted on the turn table and another one for the fixed reflectors.

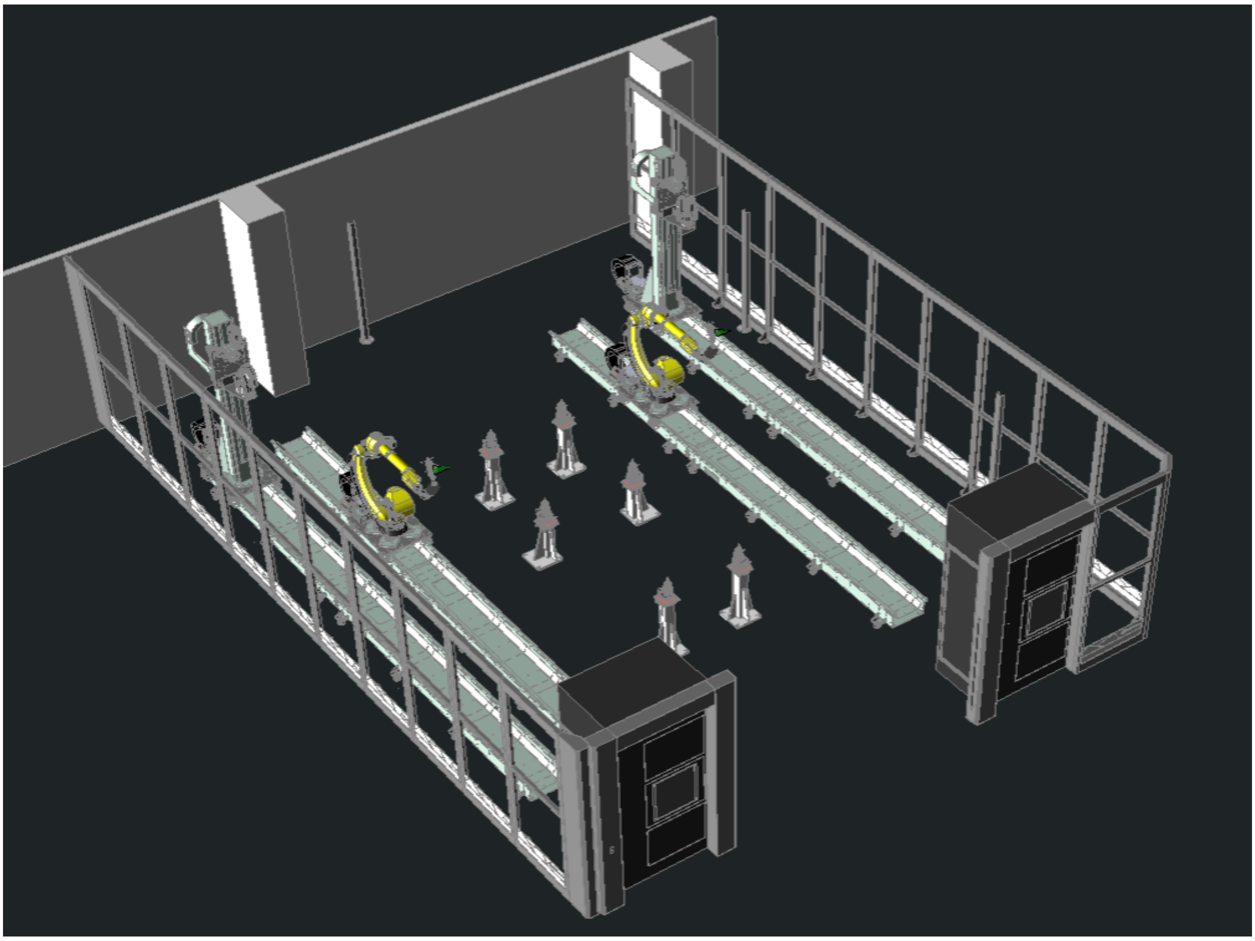
In this case whenever the turn table moves, it is needed to measure the actual positions of the relative reflector group, and whenever the tracker moves, it is needed to measure the actual positions of the fixed reflector group.

The user can refer to **Figure 8-6** keeping in mind that the turn table and tracker slider have to be considered independently within the compensation part.

8.13.5 Dual robot cell with stationary part holder and dual movable trackers

Considering a cell with the following configuration:

- 2 movable trackers over a slider
- 2 robots
- 1 group of fixed reflectors



A cell involving two robots and two trackers requires specific consideration regarding compensation, concurrent tracker usage by the two robots, collision avoidance prevention and parallel program execution capabilities to use the cell at its best capacity reducing at the same time the safety risks which are explained in the following sections. After the concepts are introduced separately, the complete workflow needed for a dual robot cell is explained in a dedicated section.

8.13.5.1 Dual tracker compensation

For dual trackers compensation the same compensation principles apply. The only other difference is how to compensate both trackers to each other.

The idea is to measure as many reflectors as possible in their nominal positions with e.g., Tracker A. Tracker A might need to move to multiple positions to measure all reflectors.

After this step it is needed to measure the actual position of the reflectors with Tracker B, where we at least need three of the reflectors that were measured by tracker A.

As a general compensation rule, it is always necessary to “compensate” every change in the relationship between the part and the tracker (due to tracker or turn table movement), to translate the taken measurement to the global coordinate system.

This means that after each tracker or turn table movement a **Measure Actual Reflectors** command must be added for the tracker we want to use in the measuring command. A **Measure Actual Reflectors** command by itself is meaningless without a **Measure Nominal Reflectors** command, so if it is needed to the move the tracker or the turn table, it’s mandatory to add a “**Measure Nominal Reflectors** command at the beginning of the program.

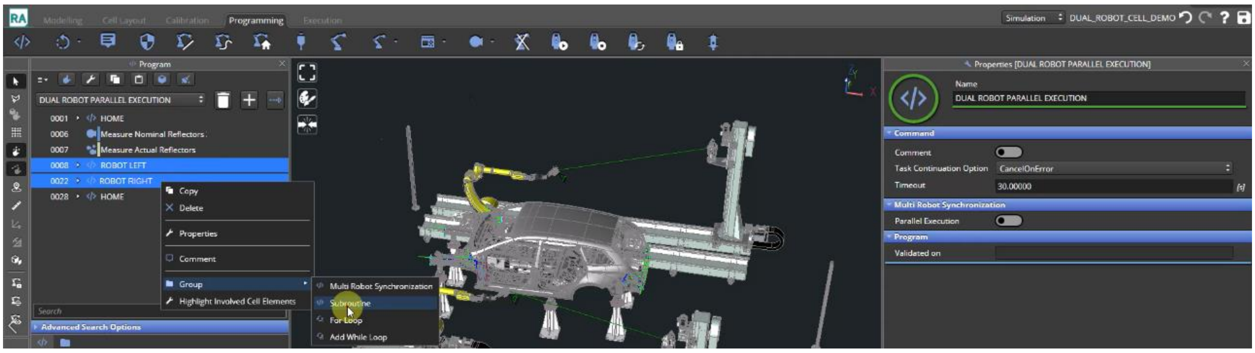
Therefore, if two trackers are available and both must be used, it’s needed to add at least a **Measure Nominal Reflectors** command for the first tracker and a **Measure Actual Reflectors** command for the second, even in case they did not move physically.

8.13.5.2 Parallel programming

To execute commands in parallel user must group those commands by **Subroutine** commands and enable the **Parallel Execution** toggle switch.

First the user have to create two separate robot subroutines (ROBOT LEFT and ROBOT RIGHT) were the two robots scan the features they need to measure. After that, the two subroutines can be grouped within a new subroutine by selecting them, right-clicking and selecting *Group* | *Subroutine*.

In this example the new subroutine is renamed “PARALLEL EXECUTION” inside its properties and is marked as parallel by enabling the **Parallel Execution** toggle switch. This means that its children (“ROBOT LEFT” and “ROBOT RIGHT” subroutines) will be executed in parallel.



DUAL ROBOT PARALLEL EXECUTION

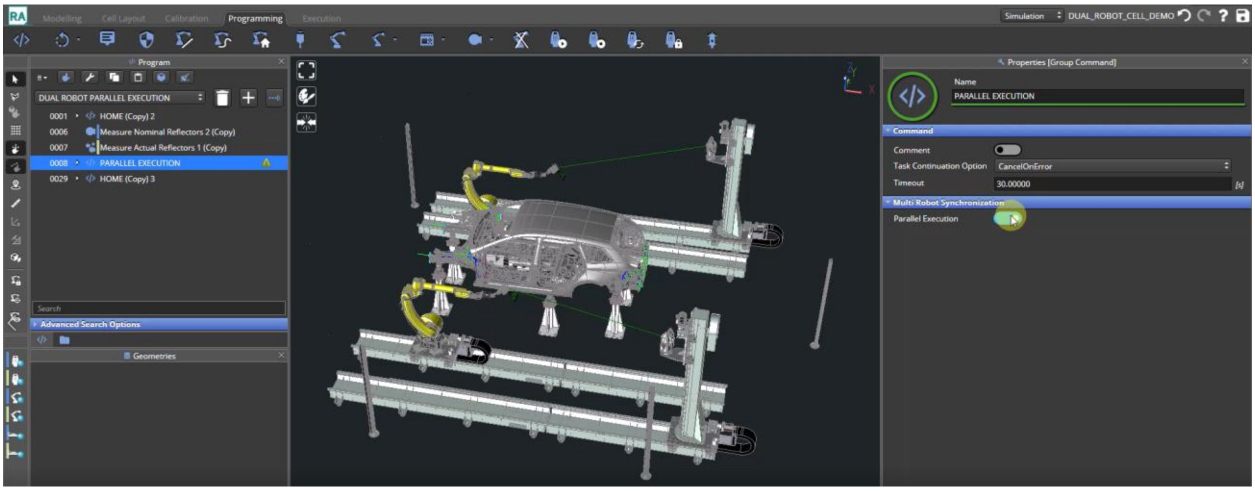
0001 HOME
0006 Measure Nominal Reflectors
0007 Measure Actual Reflectors
0008 ROBOT LEFT
0022 ROBOT RIGHT
0028 HOME

Name: DUAL ROBOT PARALLEL EXECUTION

Command
Comment ○
Task Continuation Option: CancelOnError
Timeout: 30.00000 [s]

Multi Robot Synchronization
Parallel Execution ○

Program
Validated on



DUAL ROBOT PARALLEL EXECUTION

0001 HOME (Copy) 2
0006 Measure Nominal Reflectors 2 (Copy)
0007 Measure Actual Reflectors 1 (Copy)
0008 PARALLEL EXECUTION
0029 HOME (Copy) 3

Name: PARALLEL EXECUTION

Command
Comment ○
Task Continuation Option: CancelOnError
Timeout: 30.00000 [s]

Multi Robot Synchronization
Parallel Execution ●

8.13.5.3 Tracker crossing

There might be situations where a robot cannot be used with the tracker available on the same side of the cell, e.g. Robot A and Tracker A. For this reason, it is necessary to use the tracker available on the other side of the cell and force it to lock on the scanner used for the robot in use, e.g. Robot A and Tracker B locked on Robot A scanner. The cell must be configured to use the Automation Switch Controller (ASC) and HxGN Robotic Automation allows to interact with the ASC using PLC commands as described in chapter 8.3.5.
The PLC commands are used to perform a “crossing” between the tracker and the robot as described below.

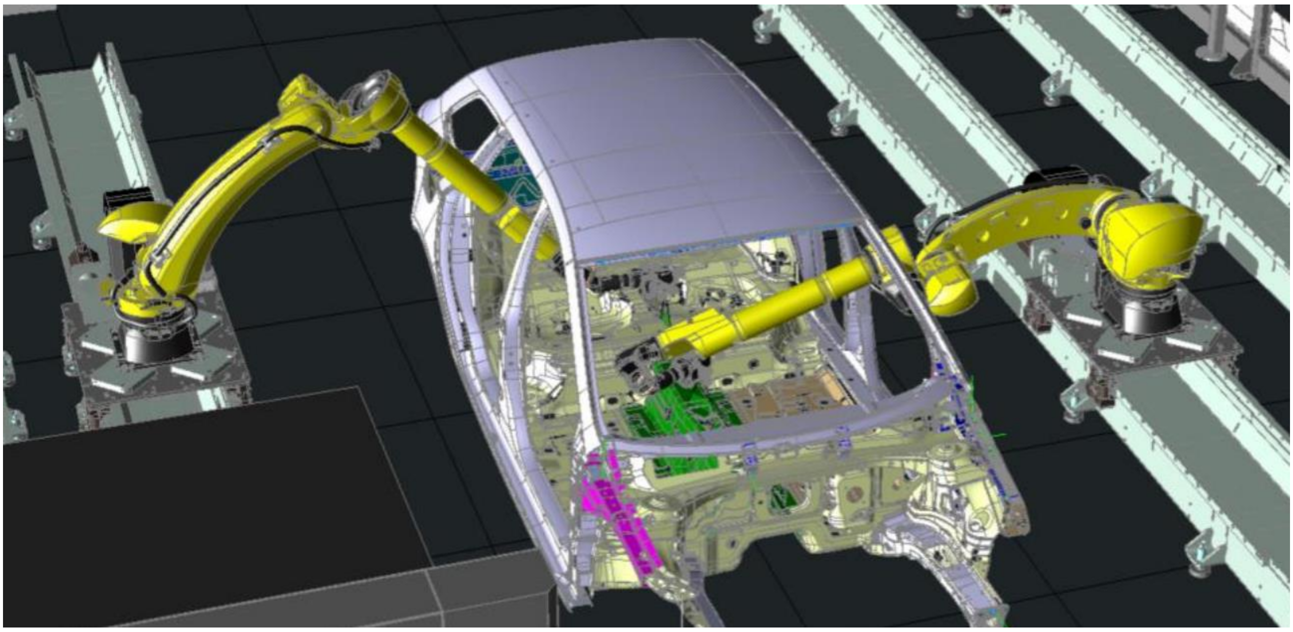
Crossing – ASC

- Add PLC Write Digital Output CMD at the start of the program to force desired consideration (not relay in theoretic status)
- Add PLC Write Digital Output CMD at the end of the program to leave the cell in the usual configuration (to avoid issues because tracker is recognizing scanner as reflector instead of AS1).

8.13.5.4 Concurrent resource

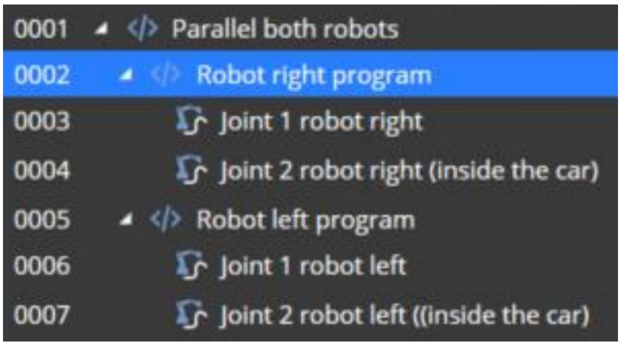
Although running robots in parallel saves overall measurement cycle time, there may be cases where it is needed to execute their movement in sequence, e.g., postpone one of them until the second has finished its task to prevent collisions.

There might be dangerous situations where both robots running in parallel are too close to each other and may collide, like in the picture below so it is paramount that when robot A is working in an interference zone, the other robot B waits for robot A to complete its task within the interference zone before entering it. For this reason, a special command is available as described in chapter 8.3.7.



Example – Lock resource to prevent collisions

Suppose we have program like the following:

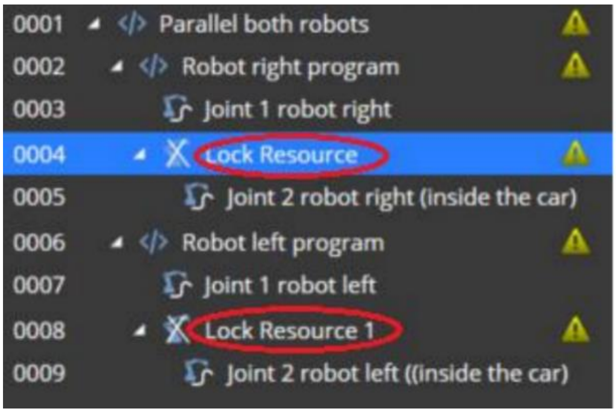


```
0001  Parallel both robots
0002    Robot right program
0003      Joint 1 robot right
0004      Joint 2 robot right (inside the car)
0005    Robot left program
0006      Joint 1 robot left
0007      Joint 2 robot left ((inside the car)
```

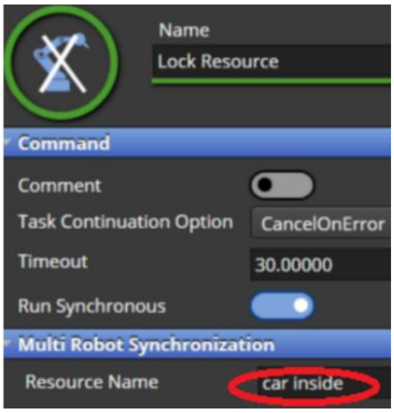
Here programs for both robots (Robot right program, Robot left program) have to be executed in parallel, but “Joint 2” motions of both are happening inside the BIW and potentially colliding.
So the potentially dangerous commands must be enclosed within **Lock Resource** commands and give them the same **Resource Name**.

Important: resource names for both commands must match exactly, e.g., “car inside” (as shown in picture below).

Now, the first robot reaching Lock Resource command will acquire the resource (“car inside”) and the second will wait until the first one will release the resource and only then continue with its execution.



```
0001  Parallel both robots
0002    Robot right program
0003      Joint 1 robot right
0004      Lock Resource
0005        Joint 2 robot right (inside the car)
0006    Robot left program
0007      Joint 1 robot left
0008      Lock Resource 1
0009        Joint 2 robot left ((inside the car)
```



Name: Lock Resource

Command

Comment ○

Task Continuation Option: CancelOnError

Timeout: 30.00000

Run Synchronous ●

Multi Robot Synchronization

Resource Name: car inside

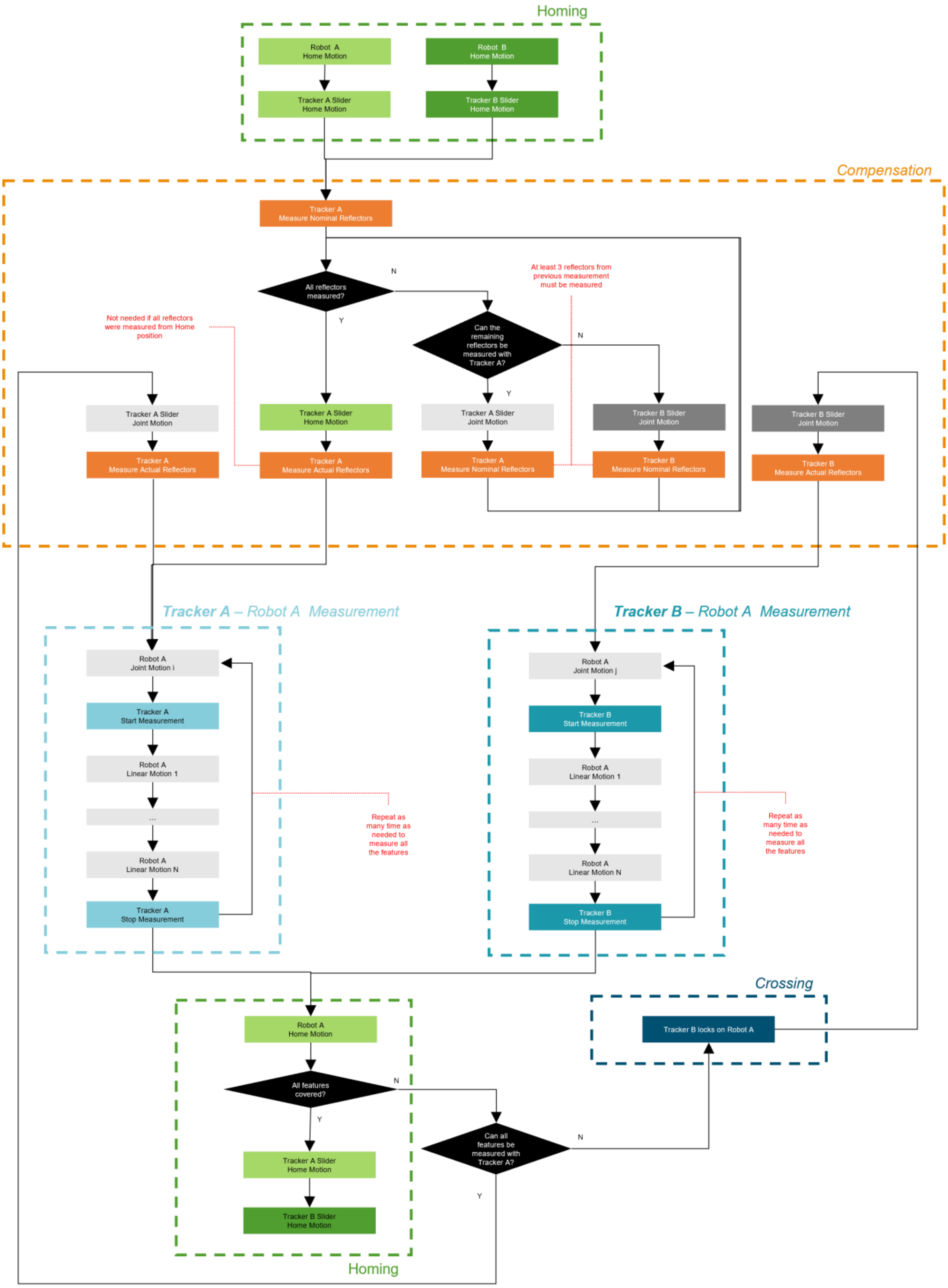
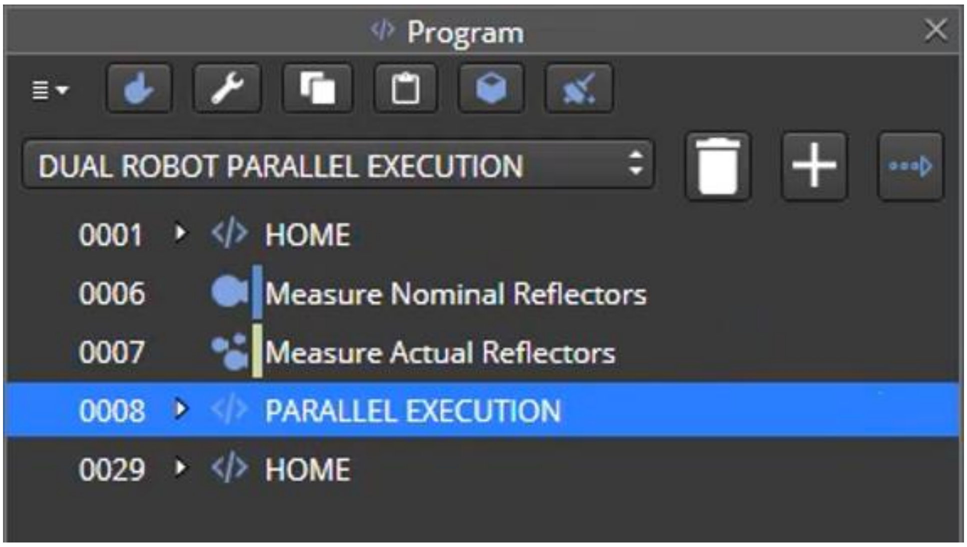

Homing
Robot A Home Motion
Robot B Home Motion
Tracker A Slider Home Motion
Tracker B Slider Home Motion
Compensation
Tracker A Measure Nominal Reflectors
All reflectors measured?
N
Y
At least 3 reflectors from previous measurement must be measured
Can the remaining reflectors be measured with Tracker A?
N
Y
Not needed if all reflectors were measured from Home position
Tracker A Slider Joint Motion
Tracker A Measure Actual Reflectors
Tracker A Slider Home Motion
Tracker A Measure Actual Reflectors
Tracker A Slider Joint Motion
Tracker A Measure Nominal Reflectors
Tracker B Slider Joint Motion
Tracker B Measure Nominal Reflectors
Tracker B Slider Joint Motion
Tracker B Measure Actual Reflectors
Tracker A – Robot A Measurement
Robot A Joint Motion i
Tracker A Start Measurement
Robot A Linear Motion 1
...
Robot A Linear Motion N
Tracker A Stop Measurement
Repeat as many time as needed to measure all the features
Tracker B – Robot A Measurement
Robot A Joint Motion j
Tracker B Start Measurement
Robot A Linear Motion 1
...
Robot A Linear Motion N
Tracker B Stop Measurement
Repeat as many time as needed to measure all the features
Crossing
Tracker B locks on Robot A
Robot A Home Motion
All features covered?
N
Y
Can all features be measured with Tracker A?
N
Y
Tracker A Slider Home Motion
Tracker B Slider Home Motion
Homing

***Figure 8-7**: Dual robot cell programming workflow using one robot and two trackers*

8.13.5.5 Complete programming workflow

The general program workflow for a parallel execution program is described in **Figure 8-8** and contain the following main steps:

1) Send all robots and sliders to “Home” position (may be in parallel) as depicted in **Figure 8-7** and **Figure 8-8**.



0001 HOME
0006 Measure Nominal Reflectors
0007 Measure Actual Reflectors
0008 PARALLEL EXECUTION
0029 HOME

2) Perform necessary compensation steps as explained in chapter 8.13.5.1.

3) Create sub-routines as separate programs for the two Robot A / Tracker A and Robot B / Tracker B pairs in parallel as described in chapter 8.13.5.2. This step may be repeated as many times as needed to measure all the features.

4) There might be situations when both robots running in parallel are only allowed to use the same tracker (the second is unreachable). A tracker can lock only on a face of the scanner mounted on the robot. E.g. when Tracker A is locked on a face of Robot A scanner, Robot B must wait for Robot A to complete its task, before Tracker A could lock on a face of Robot B scanner.
For this situation the tracker crossing functionality is needed as described in chapter 8.13.5.3.
The flowchart depicted in **Figure 8-7** shows a workflow that includes only the measurement parts using Robot A together with either Tracker A or Tracker B.
A workflow containing both robot measurement routines have two additional measurement branches relative to **Tracker B – Robot B** and **Tracker A – Robot B** which are exactly the same as the one described in Robot A measurement branches as depicted in **Figure 8-8**.

5) Whenever the measurement branches of Robot A and Robot B must inspect an interference zone, the concurrent resource usage functionality must be used as described in chapter 8.13.5.4 and depicted in **Figure 8-8**.

6) Send all robots and sliders to “Home” position (may be in parallel) as depicted in **Figure 8-7** and **Figure 8-8**.

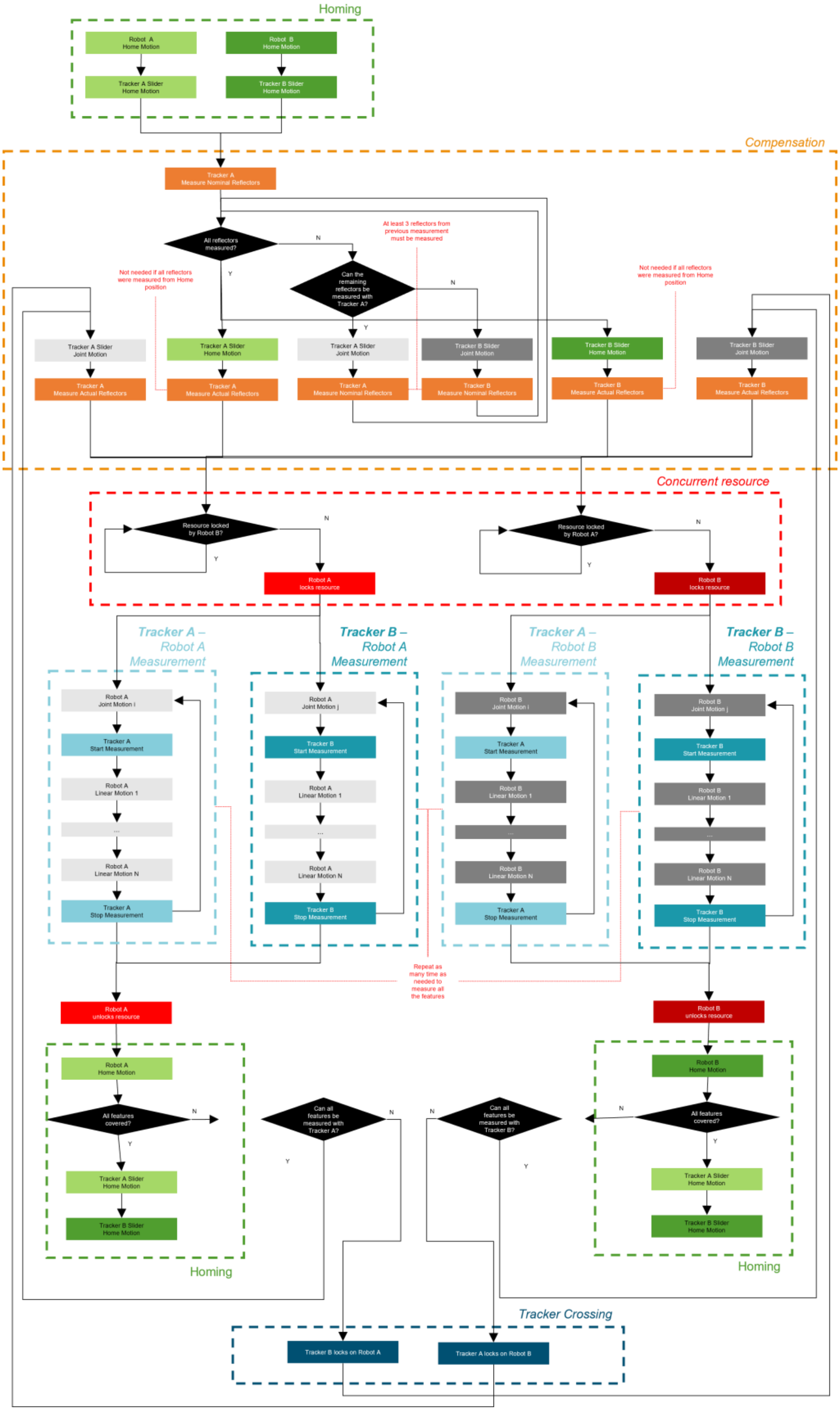

Homing
Robot A Home Motion
Robot B Home Motion
Tracker A Slider Home Motion
Tracker B Slider Home Motion
Compensation
Tracker A Measure Nominal Reflectors
All reflectors measured?
N
Y
At least 3 reflectors from previous measurement must be measured
Can the remaining reflectors be measured with Tracker A?
N
Y
Not needed if all reflectors were measured from Home position
Not needed if all reflectors were measured from Home position
Tracker A Slider Joint Motion
Tracker A Slider Home Motion
Tracker A Slider Joint Motion
Tracker B Slider Joint Motion
Tracker B Slider Home Motion
Tracker B Slider Joint Motion
Tracker A Measure Actual Reflectors
Tracker A Measure Actual Reflectors
Tracker A Measure Nominal Reflectors
Tracker B Measure Nominal Reflectors
Tracker B Measure Actual Reflectors
Tracker B Measure Actual Reflectors
Concurrent resource
Resource locked by Robot B?
N
Y
Robot A locks resource
Resource locked by Robot A?
N
Y
Robot B locks resource
Tracker A – Robot A Measurement
Tracker B – Robot A Measurement
Tracker A – Robot B Measurement
Tracker B – Robot B Measurement
Robot A Joint Motion i
Tracker A Start Measurement
Robot A Linear Motion 1
...
Robot A Linear Motion N
Tracker A Stop Measurement
Robot A Joint Motion j
Tracker B Start Measurement
Robot A Linear Motion 1
...
Robot A Linear Motion N
Tracker B Stop Measurement
Robot B Joint Motion i
Tracker A Start Measurement
Robot B Linear Motion 1
...
Robot B Linear Motion N
Tracker A Stop Measurement
Robot B Joint Motion j
Tracker B Start Measurement
Robot B Linear Motion 1
...
Robot B Linear Motion N
Tracker B Stop Measurement
Repeat as many time as needed to measure all the features
Robot A unlocks resource
Robot B unlocks resource
Robot A Home Motion
All features covered?
N
Y
Tracker A Slider Home Motion
Tracker B Slider Home Motion
Homing
Can all features be measured with Tracker A?
N
Y
Can all features be measured with Tracker B?
N
Y
Robot B Home Motion
All features covered?
N
Y
Tracker A Slider Home Motion
Tracker B Slider Home Motion
Homing
Tracker Crossing
Tracker B locks on Robot A
Tracker A locks on Robot B

***Figure 8-8**: Complete dual robot cell programming workflow using two robots and two trackers*

9 Execution

Execution is the phase where a program that has been generated previously in Programming mode is run to perform a measurement task on a part.

Once a program is generated the user has the option to either run it in the Execution mode inside HxGN Robotic Automation by selecting the relative tab or run it outside of HxGN Robotic Automation within the Execution Manager, a web application specifically developed to running on a browser in kiosk mode for easy execution of programs by cell operators.

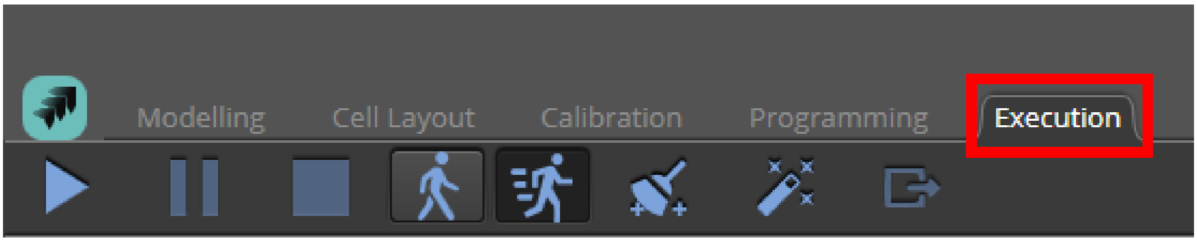
The user can run the program in both real and simulation environments. In real environment to execute a program, all the relative real cell components in use must be connected either to HxGN Robotic Automation or Execution Manager.

Warning: before using a program within the Execution Manager, the user must validate the program and run it at least once in an online environment with HxGN Robotic Automation in Execution mode.

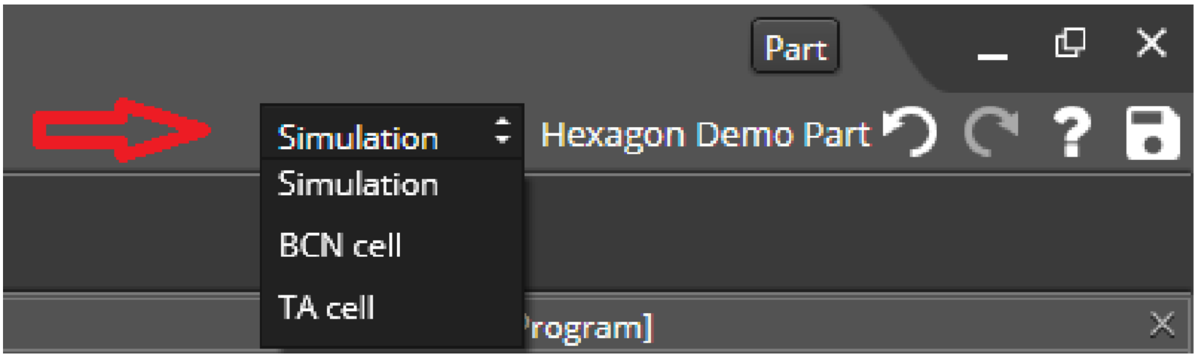
Warning: when using the “Move Machine” functionality in the real environment make sure all the other machines are in their correct position as it may cause collisions.

9.1 Execution from HxGN Robotic Automation

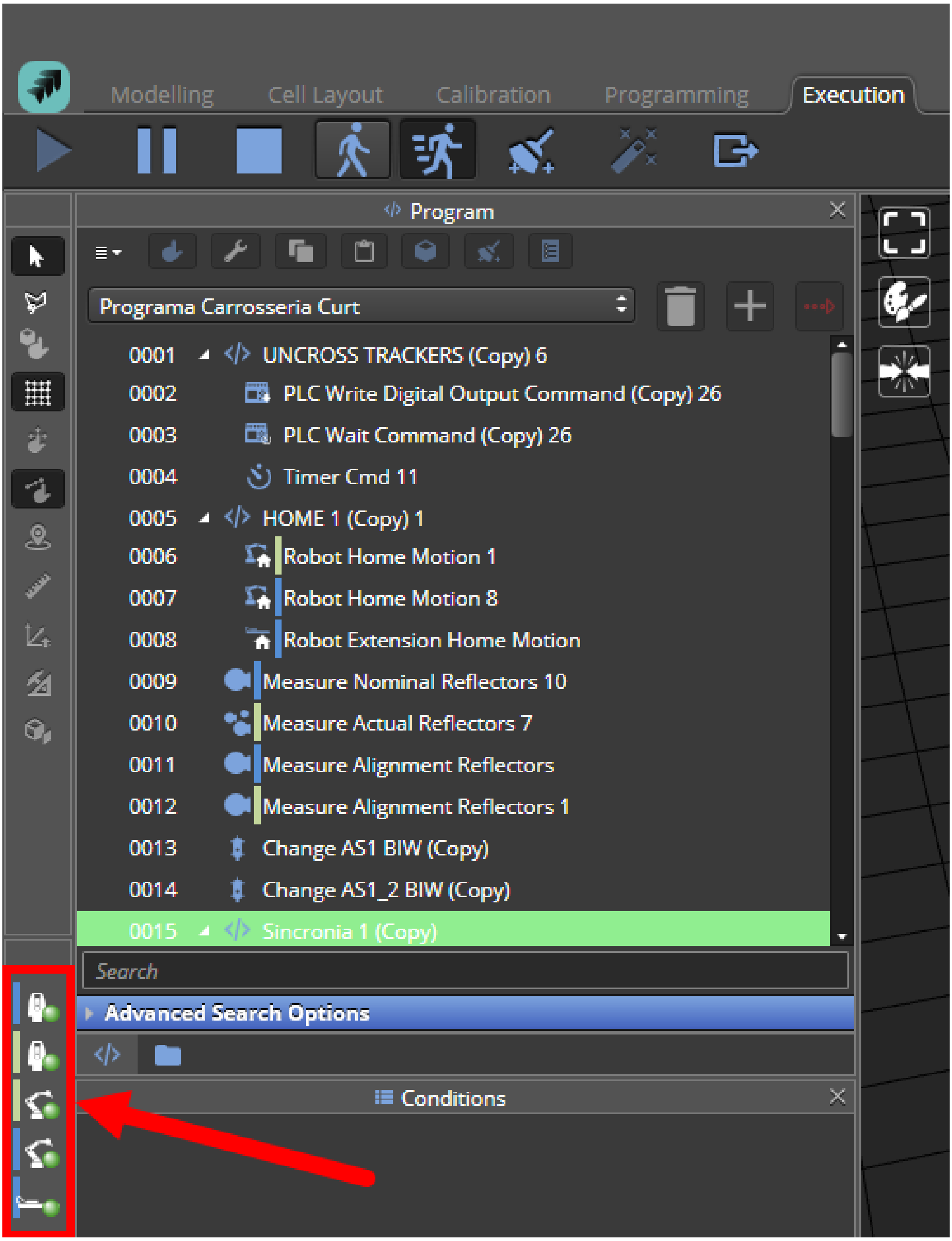
The execution of a program in HxGN Robotic Automation is managed within the **Execution** tab.



User can execute the program either in Simulation mode or in a Real environment i.e., connected to moving hardware in real. It is possible to select if using the simulation mode or to which real cell to connect to by selecting the environment in the top right part of the UI via a drop-down menu.



The configuration of cell name is done in the cell .json file creating a “Cell.json” file where “Cell” is the name of that cell model, e.g., in the example above “BCN cell.json” or “TA cell.json”.
Once the cell is connected to the real cell, the relative connection buttons will turn from blue to green.



Programa Carrosseria Curt

0001 UNCROSS TRACKERS (Copy) 6
0002 PLC Write Digital Output Command (Copy) 26
0003 PLC Wait Command (Copy) 26
0004 Timer Cmd 11
0005 HOME 1 (Copy) 1
0006 Robot Home Motion 1
0007 Robot Home Motion 8
0008 Robot Extension Home Motion
0009 Measure Nominal Reflectors 10
0010 Measure Actual Reflectors 7
0011 Measure Alignment Reflectors
0012 Measure Alignment Reflectors 1
0013 Change AS1 BIW (Copy)
0014 Change AS1_2 BIW (Copy)
0015 Sincronia 1 (Copy)

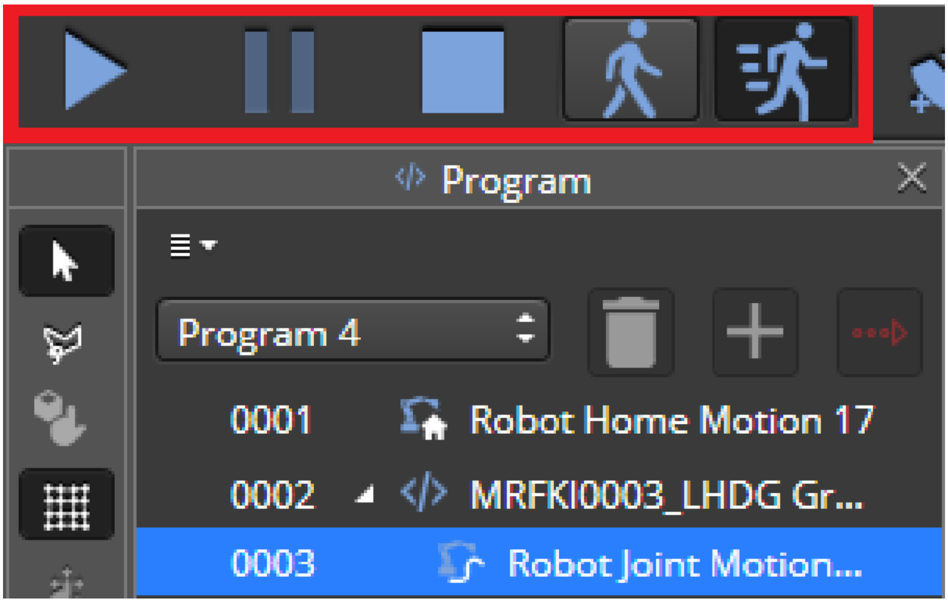
The user can choose between two modes of running the program:

1) **Step Through** (🚶): running the program step by step stopping after each of the commands and pressing the **Start** button to continue.

2) **Run** (🏃): running the program without any stops transitioning automatically between the commands.

To run certain commands of the program the user can either select a specific command and execute the program from there pressing the **Start** button (▶) or first click the **Start** button and then select the command to be executed first.

The user can at any moment decide to click **Pause** button (⏸) to interrupt the program at the end of the current command or pressing **Stop** button (■) to abort the program altogether.



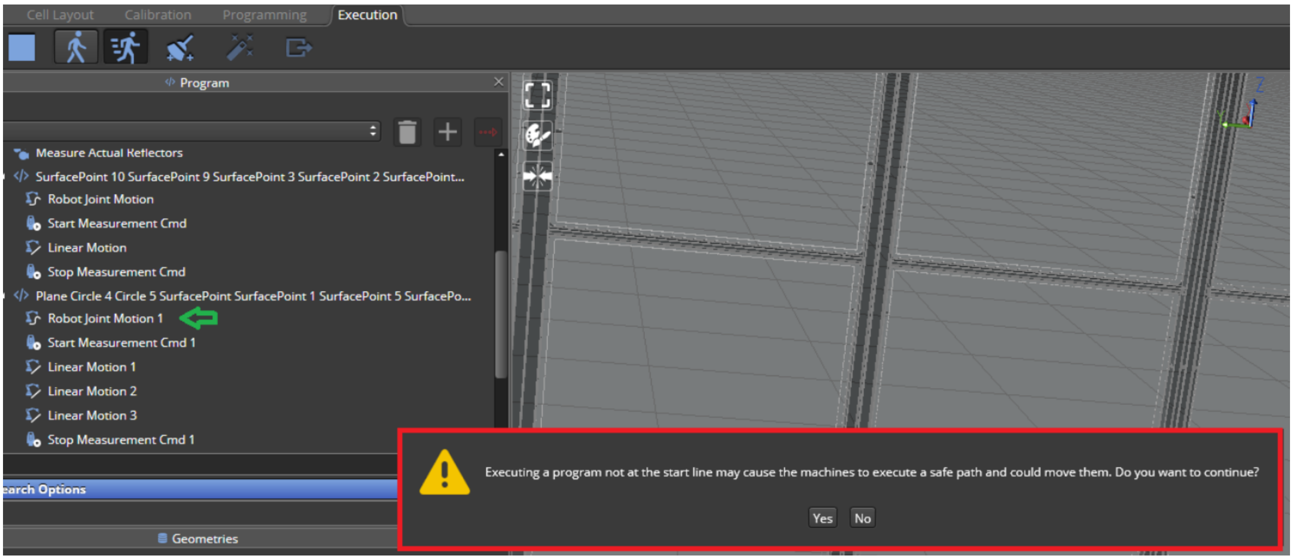
0001 Robot Home Motion 17

0002 MRFKI0003_LHDG Gr...

0003 Robot Joint Motion...

***Figure 9-1**: “Start”, “Pause”, “Stop”, “Step through”, and “Run” buttons a with a selected command from the program.*

When trying to execute a program not from the beginning, the following message will appear:

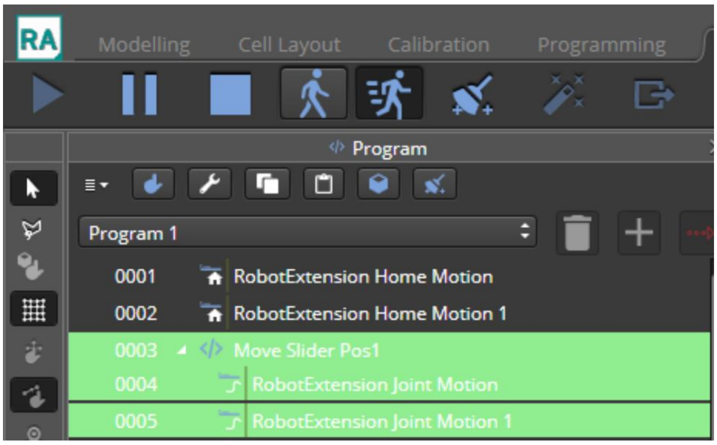


Measure Actual Reflectors
SurfacePoint 10 SurfacePoint 9 SurfacePoint 3 SurfacePoint 2 SurfacePoint...
Robot Joint Motion
Start Measurement Cmd
Linear Motion
Stop Measurement Cmd
Plane Circle 4 Circle 5 SurfacePoint SurfacePoint 1 SurfacePoint 5 SurfacePo...
Robot Joint Motion 1
Start Measurement Cmd 1
Linear Motion 1
Linear Motion 2
Linear Motion 3
Stop Measurement Cmd 1

Executing a program not at the start line may cause the machines to execute a safe path and could move them. Do you want to continue?

If the user clicks **Yes**, HxGN Robotic Automation will compute a new safe path from the current position to the selected command. If the user clicks **No**, the execution will be aborted.

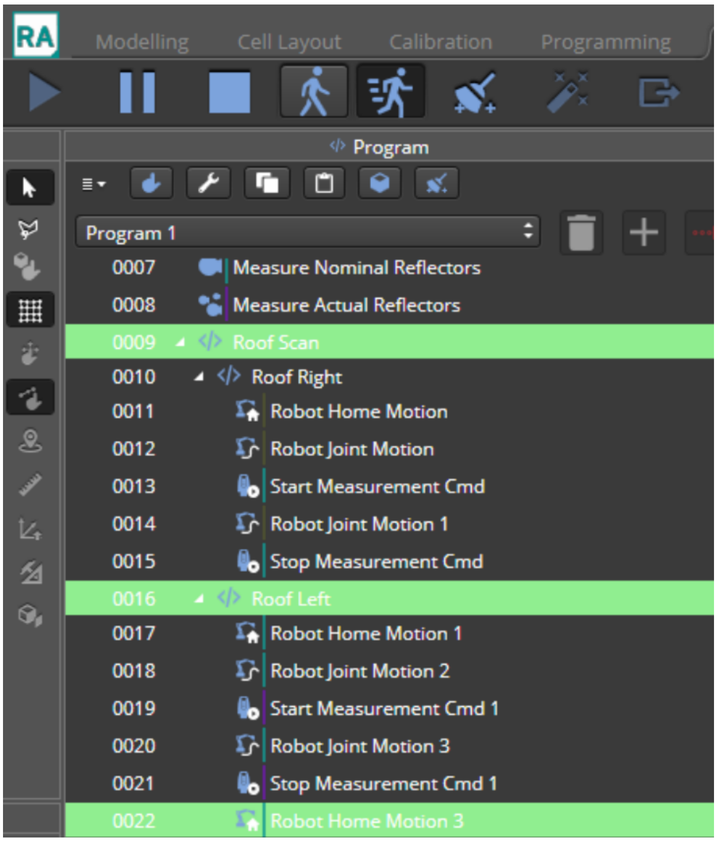
When a program is executed, the command line being executed is highlighted in green.
If a command is highlighted, it's parent will be highlighted as well.
If the program has an error, the last active command will stay highlighted.



Program 1

0001 RobotExtension Home Motion
0002 RobotExtension Home Motion 1
0003 Move Slider Pos1
0004 RobotExtension Joint Motion
0005 RobotExtension Joint Motion 1

Figure 9-2*: example of 2 commands run at the same time. Commands are highlighted and the subroutine parent folder.*

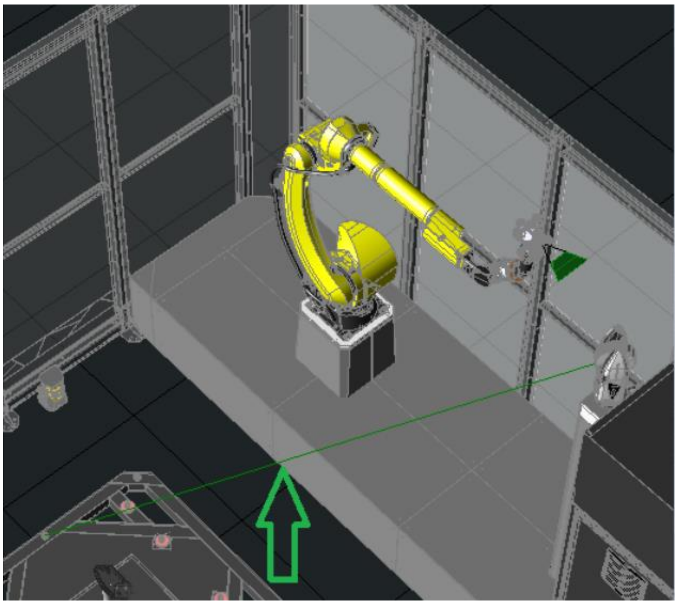


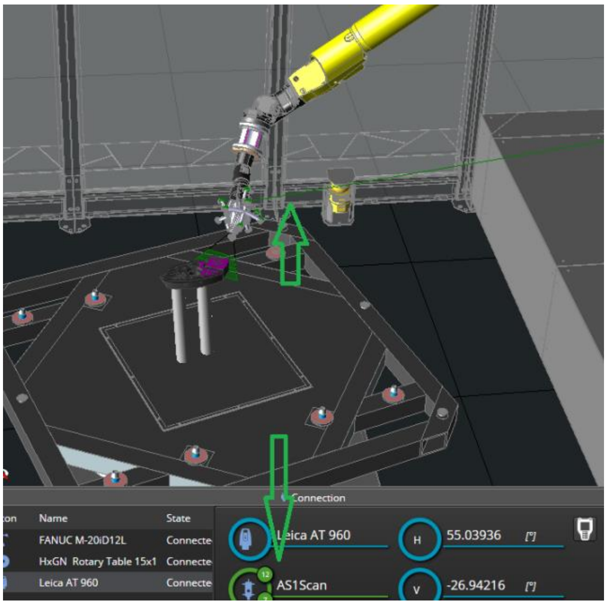
Program 1

0007 Measure Nominal Reflectors
0008 Measure Actual Reflectors
0009 Roof Scan
0010 Roof Right
0011 Robot Home Motion
0012 Robot Joint Motion
0013 Start Measurement Cmd
0014 Robot Joint Motion 1
0015 Stop Measurement Cmd
0016 Roof Left
0017 Robot Home Motion 1
0018 Robot Joint Motion 2
0019 Start Measurement Cmd 1
0020 Robot Joint Motion 3
0021 Stop Measurement Cmd 1
0022 Robot Home Motion 3

Figure 9-3*: "Roof Scan" Parent is highlighted as "Roof Left" subroutine is still executing the "Robot Home Motion 3" command (and "Roof Right" subroutine is done already).*

During program execution, the laser beam will be green when it is locked on a reflector or a scanner face and it can get a 6DOF measurement.





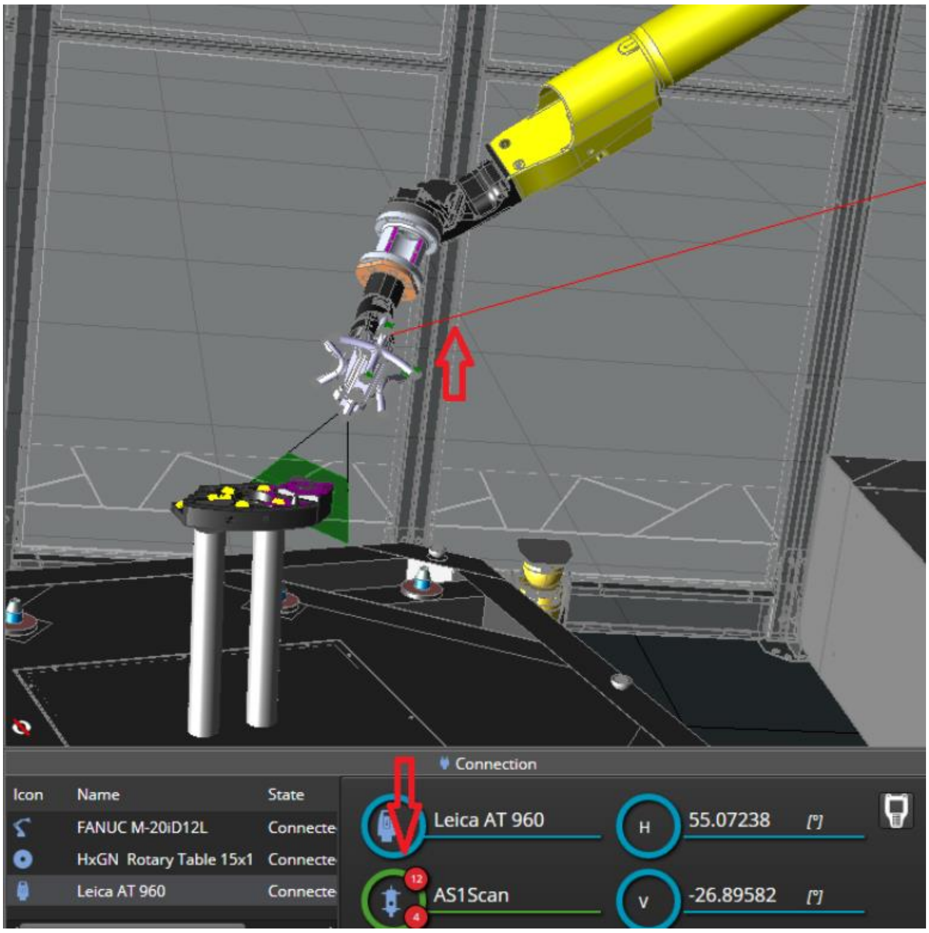
Connection

on	Name	State
	FANUC M-20iD12L	Connecte
	HxGN Rotary Table 15x1	Connecte
	Leica AT 960	Connecte

Leica AT 960 H 55.03936 [°]

AS1Scan V -26.94216 [°]

Otherwise, the laser beam will be coloured in red.



Connection

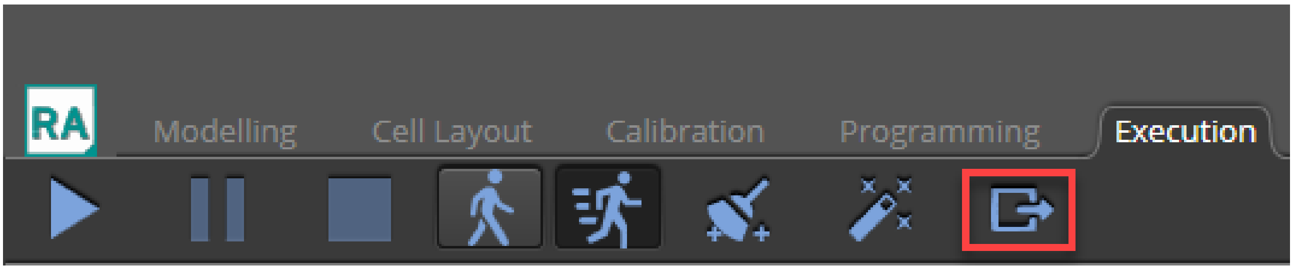
Icon	Name	State
	FANUC M-20iD12L	Connecte
	HxGN Rotary Table 15x1	Connecte
	Leica AT 960	Connecte

Leica AT 960 H 55.07238 [°]

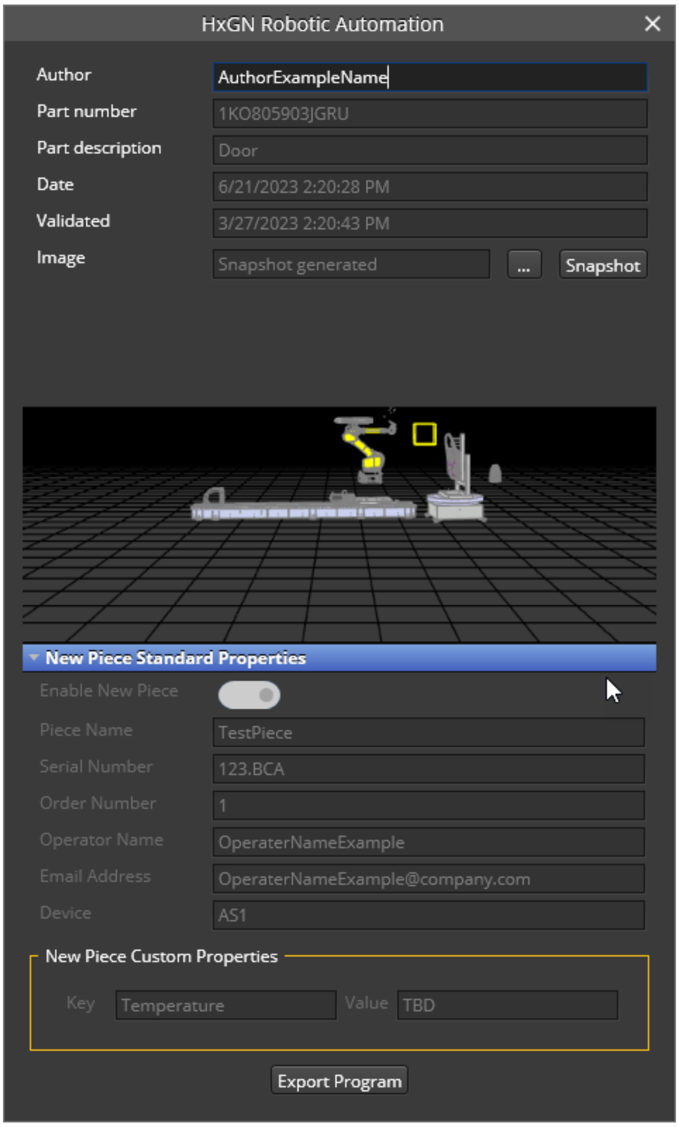
AS1Scan V -26.89582 [°]

9.2 Execution from Execution Manager (EM)

As explained at the beginning of chapter 9, before a program can be run from Execution Manager (EM) it must be validated and executed from RA in the real cell represented by its digital twin where the program was generated offline and executed in simulation.
After this execution the export button will be available.



Clicking on it, will open the export program pop-up. Filling in the required fields, will let the program to be exported correctly within the internal database.


HxGN Robotic Automation

Author	AuthorExampleName
Part number	1KO805903JGRU
Part description	Door
Date	6/21/2023 2:20:28 PM
Validated	3/27/2023 2:20:43 PM
Image	Snapshot generated

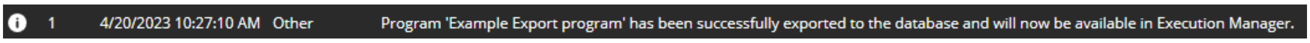
New Piece Standard Properties

Enable New Piece	
Piece Name	TestPiece
Serial Number	123.BCA
Order Number	1
Operator Name	OperaterNameExample
Email Address	OperaterNameExample@company.com
Device	AS1

New Piece Custom Properties

Key Temperature Value TBD

After clicking on *Export Program,* a successful export confirmation message will appear in the status window.

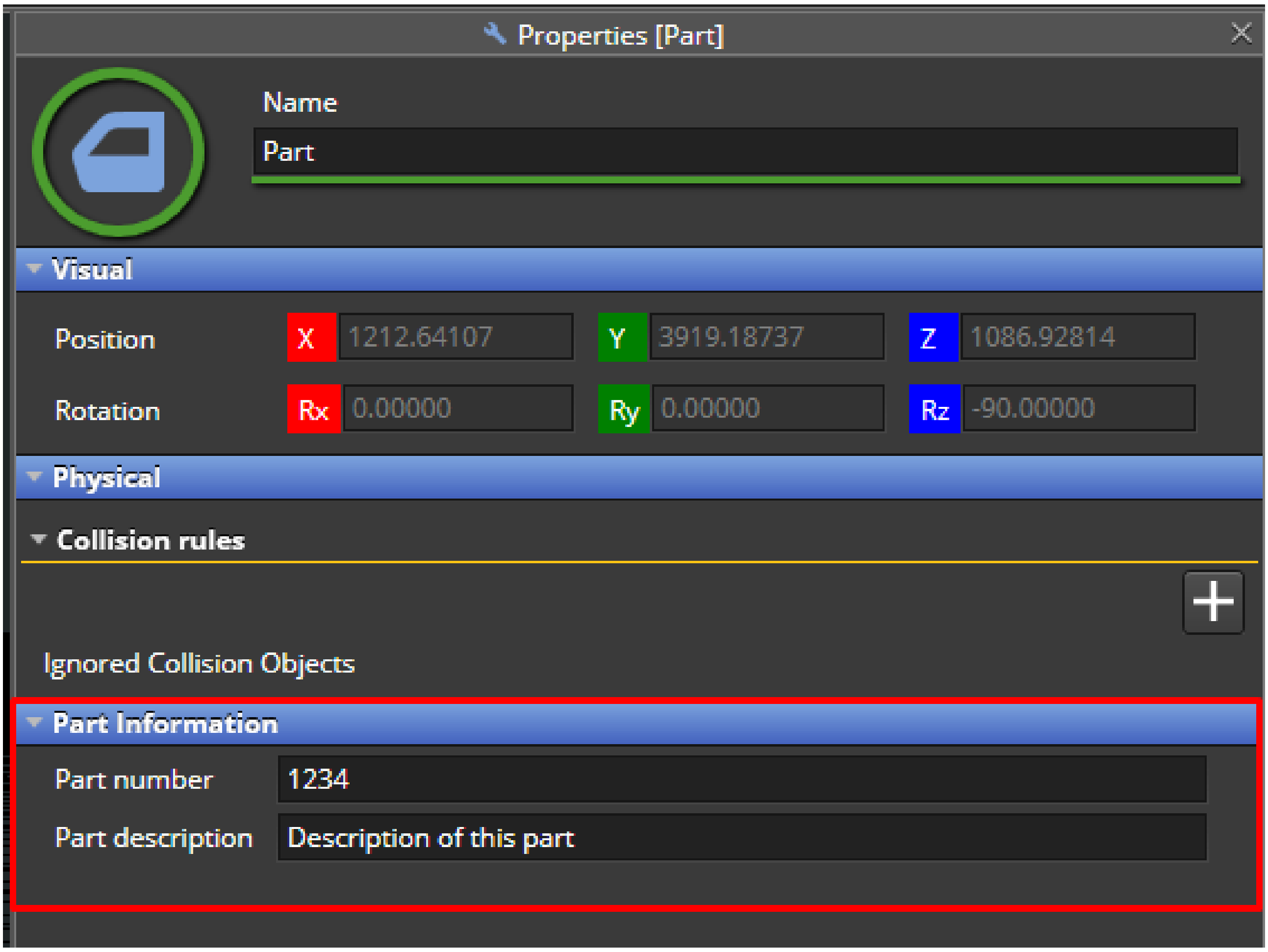


```
1   4/20/2023 10:27:10 AM   Other   Program 'Example Export program' has been successfully exported to the database and will now be available in Execution Manager.
```

Info: For the program to appear within the available programs in Execution Manager it may be needed to refresh the browser page [F5].

9.2.1 Part properties

New part number and description fields are available within Part Information fields in Properties:



Properties [Part]

Name: Part

Visual

Position	X	1212.64107	Y	3919.18737	Z	1086.92814
Rotation	Rx	0.00000	Ry	0.00000	Rz	-90.00000

Physical

Collision rules

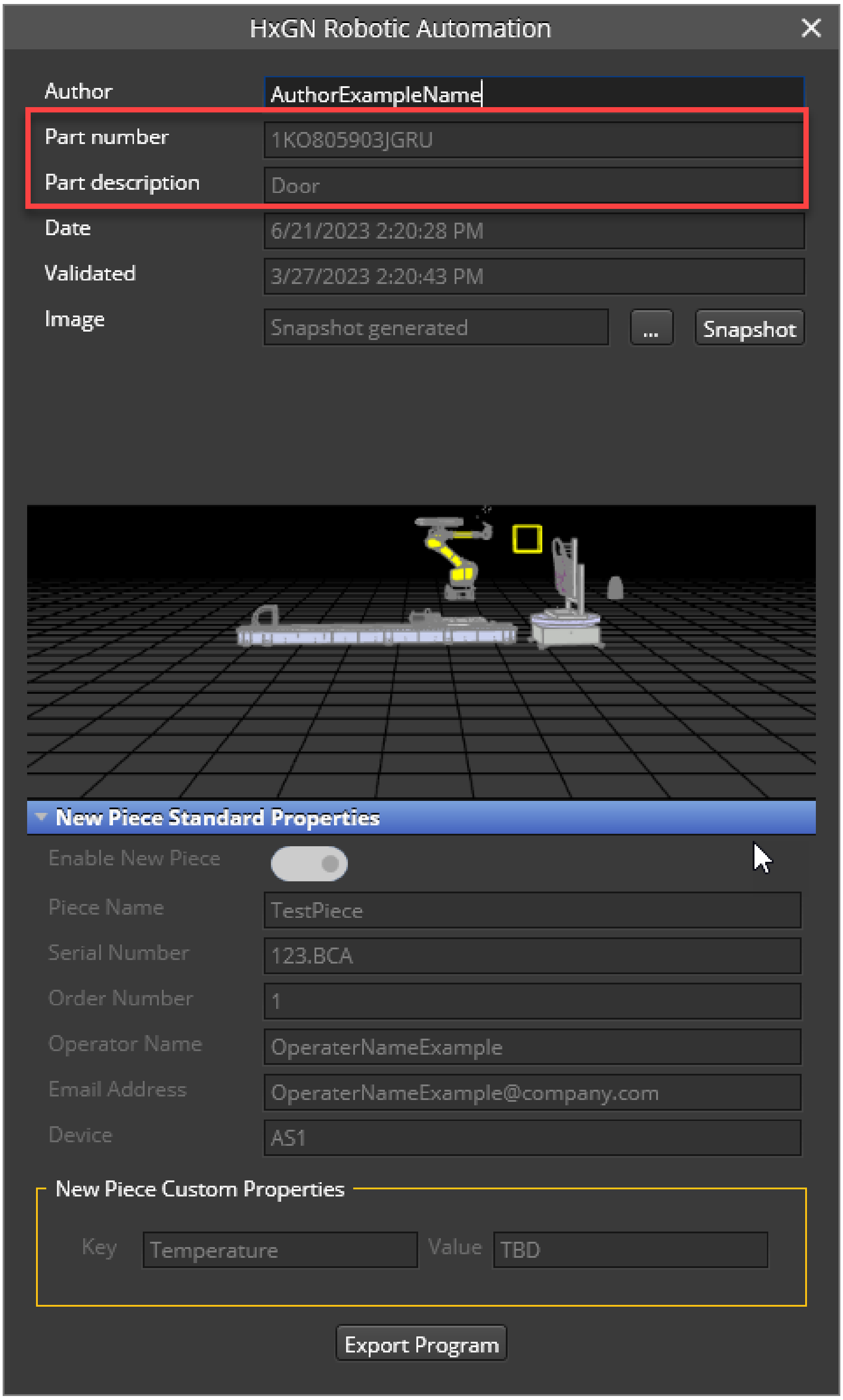
Ignored Collision Objects

Part Information

Part number: 1234

Part description: Description of this part

When the export program dialog box opens, the fields are automatically filled with the respective values defined in the part properties. In case they are not defined in the properties, they can be filled in the program export dialog box and they will be also updated and saved in the part properties.



HxGN Robotic Automation

Author	AuthorExampleName
Part number	1KO805903JGRU
Part description	Door
Date	6/21/2023 2:20:28 PM
Validated	3/27/2023 2:20:43 PM
Image	Snapshot generated

New Piece Standard Properties

Enable New Piece	
Piece Name	TestPiece
Serial Number	123.BCA
Order Number	1
Operator Name	OperaterNameExample
Email Address	OperaterNameExample@company.com
Device	AS1

New Piece Custom Properties

Key Temperature Value TBD

Metadata (also called New Piece properties in PolyWorks) can be added to any program.
In case the user renames a program in HxGN Robotic Automation, validates it, runs it in the real cell and exports it to Execution Manager, then Execution Manager will overwrite the existing program name with the new one.

9.2.2 SQL Express

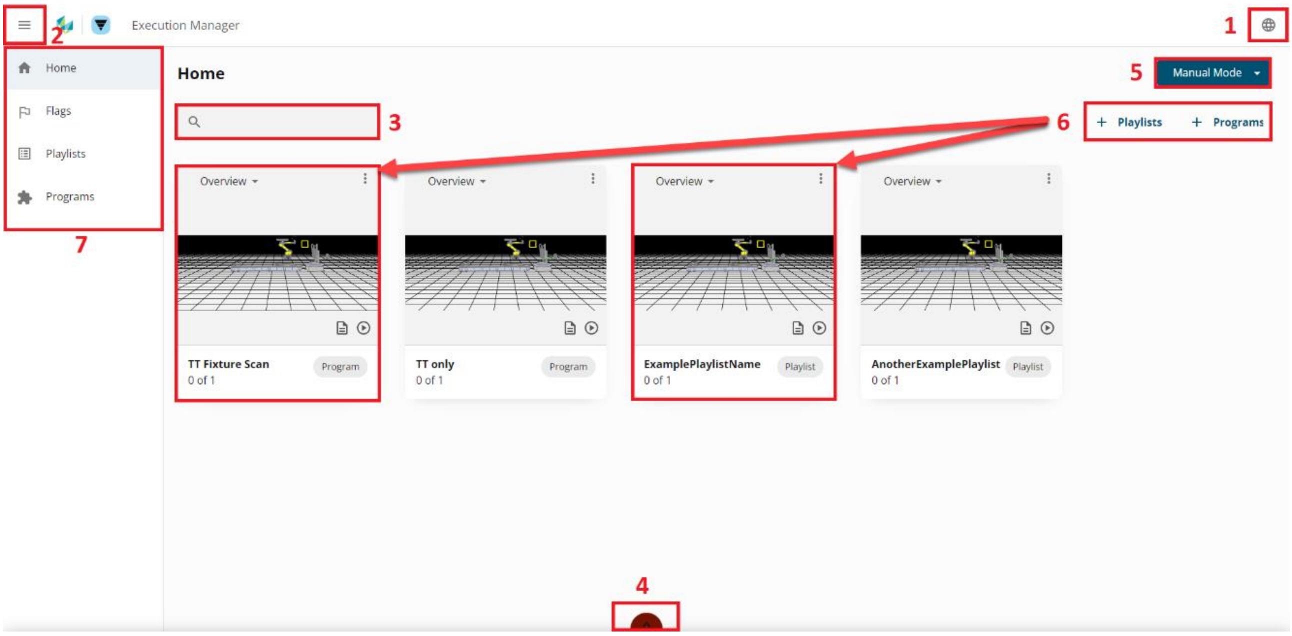
Both HxGN Robotic Automation and Execution Manager require the installation of SQL Express as an internal database where the exported programs are stored. The HxGN Robotic Automation installer has been modified so it will install SQL Express in case it is not installed yet on the system.

Warning: In case the Generic Host has issues accessing the SQL Express database. We recommend the following steps:

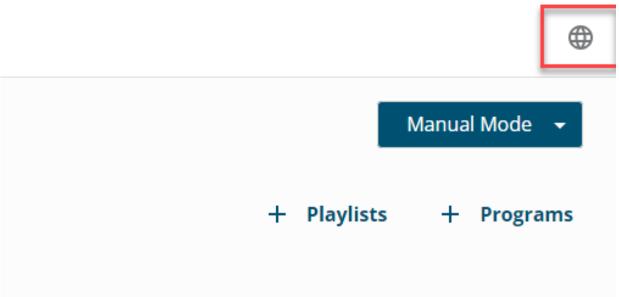
1) Stop the HxGN.RA.ServerGenericHost service
2) Check connection string in: “C:\ProgramData\Hexagon\HxGN.RA.GenericHost\DataProviderSettings.json" **"DefaultConnection": "Server=.\\SQLEXPRESS;Database=RA;Integrated Security=True;TrustServerCertificate=Yes"**
3) Run “C:\Program Files\Hexagon\ServerGenericHost\HxGN.RA.GenericHost.exe”
4) The console application will grant the permissions so RA can access the Database as a service
5) (Optional) Stop the console application
6) (Optional) Start the HxGN.RA.ServerGenericHost service

9.2.3 UI overview

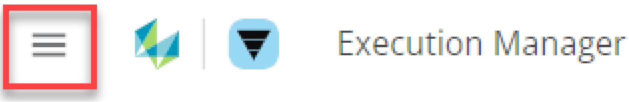
The Execution Manager UI is divided into ten principal areas as depicted in the following picture.
At the first execution the Home page is shown and pages can be navigated from the left side bar.



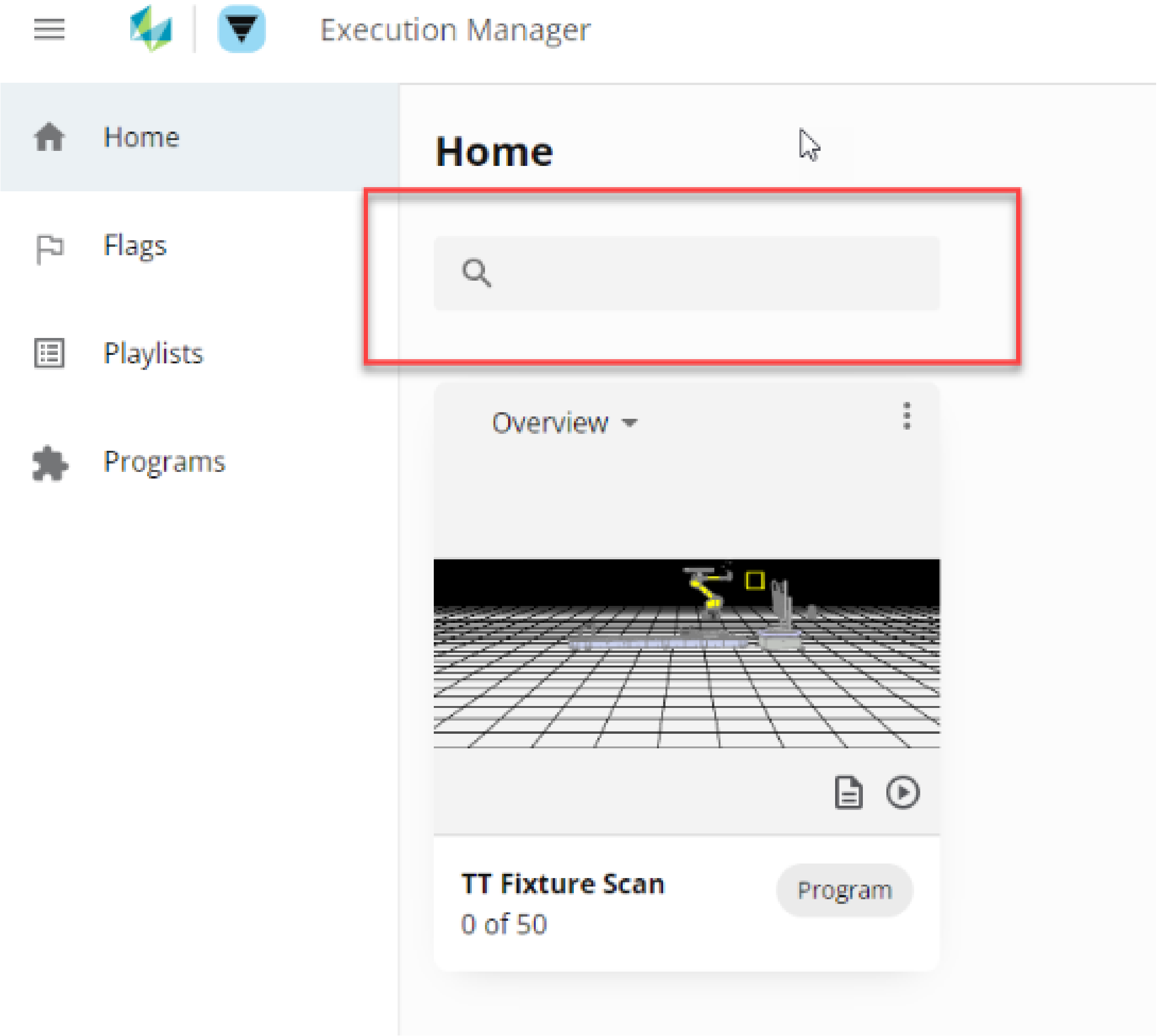
1) **Language Selection** to change the Execution Manager language.



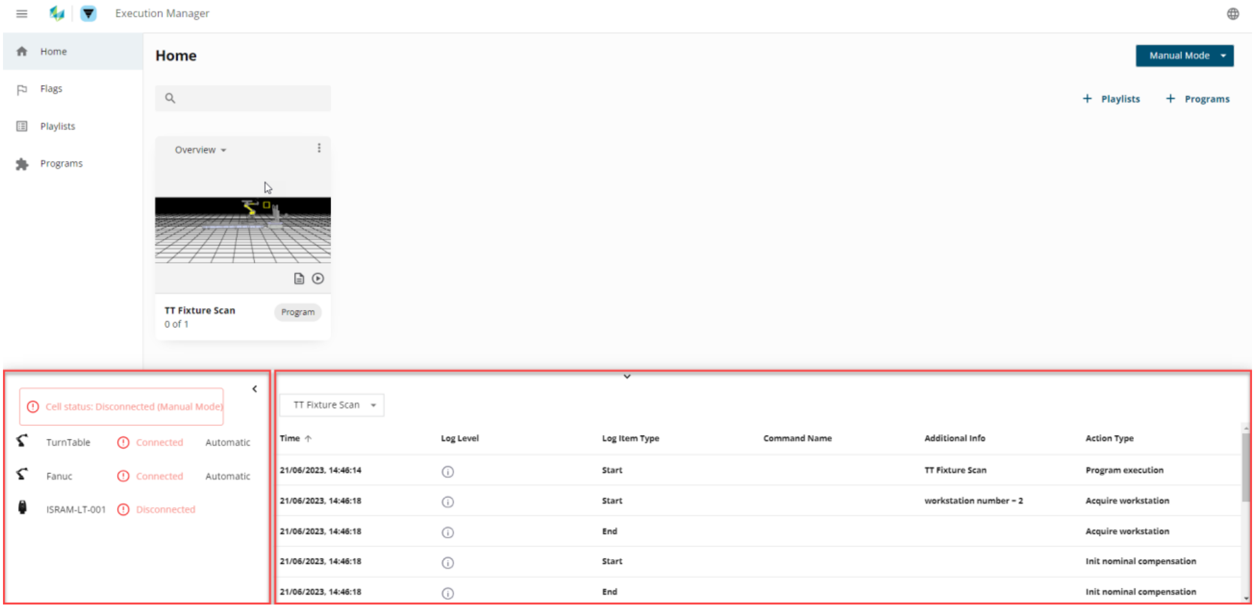
2) **Toggle Side bar** expands or collapses the side bar by clicking the burger menu.



3) **Search bar** allows the user to search and filter within the current tab by name or part number.



4) **Cell status flyout**: when clicked it shows the user a collapsible view of:
   a) the status of the cell items and the connected metrology application
   b) the logs of the last execution for each of the program/playlist cards available in the home page



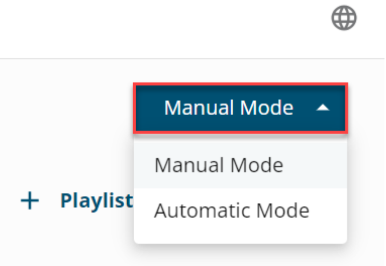
Cell status: Disconnected (Manual Mode)

TurnTable	Connected	Automatic
Fanuc	Connected	Automatic
ISRAM-LT-001	Disconnected	

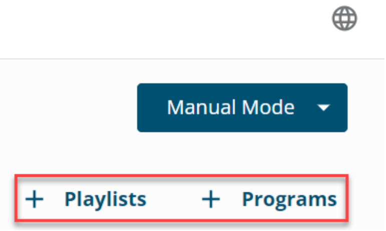
TT Fixture Scan

Time ↑	Log Level	Log Item Type	Command Name	Additional Info	Action Type
21/06/2023, 14:46:14	ⓘ	Start		TT Fixture Scan	Program execution
21/06/2023, 14:46:18	ⓘ	Start		workstation number - 2	Acquire workstation
21/06/2023, 14:46:18	ⓘ	End			Acquire workstation
21/06/2023, 14:46:18	ⓘ	Start			Init nominal compensation
21/06/2023, 14:46:18	ⓘ	End			Init nominal compensation

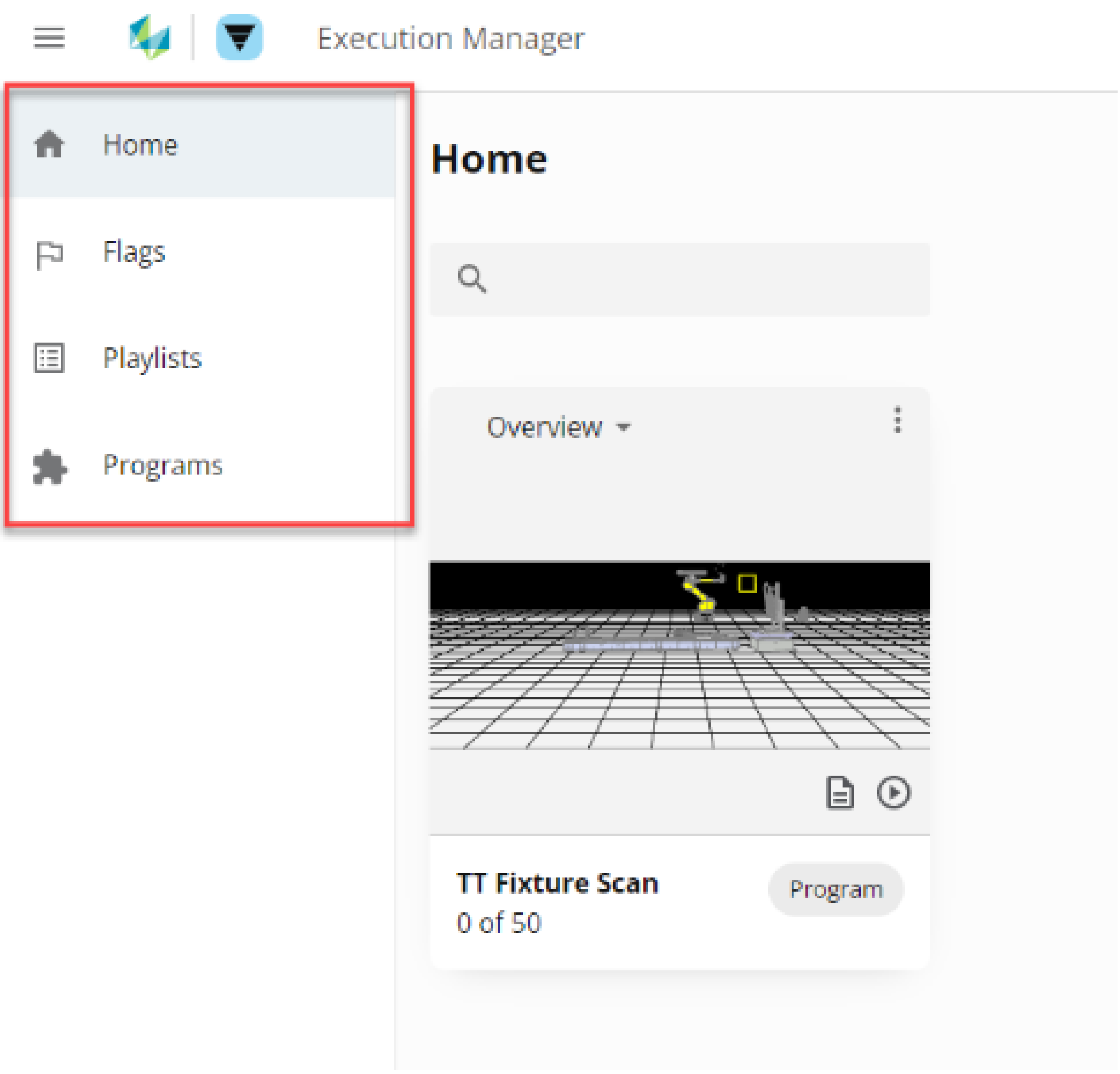
5) **Automatic/Manual mode toggle** to switch between manual or automatic modes
   a) In automatic mode the cell PLC will control the execution of the playlists by sending the relative Flag Code to the Execution Manager
   b) In manual mode programs and playlists can be executed in the order defined by the user



6) **Add a playlist** or **Add a program** to the home page.



7) **Side bar** to navigate among the following pages:
   a) **Home page:** page that allows to select and run desired program or playlists, see the cell status and switch between automatic and manual mode.
   b) **Flags page**: allows to configure the flags and associate them to playlists. Execution Manager running in automatic mode will listen to the flags set by the PLC to execute the associated playlist.
   c) **Playlist page** allows to create playlists made of existing programs.
   d) **Program page** shows a list of all the programs in the database and allow the user to import programs previously exported with HxGN Robotic Automation version 2.1.2 and newer. Additionally, the user can edit the metadata of a program.



9.2.4 General User workflow

Before start using the Execution Manager, ensure that is possible to communicate on port 5000 and that HxGN.RA.ServerGenericHost service is up and running.
If both conditions hold it is possible to navigate to http://localhost:5000/.

Execution Manager is a kiosk interface aimed to be used by cell operators to easily execute programs generated with HxGN Robotic Automation and create playlists composed of such programs.

A program can be:

- Added directly to the home page to be executed
- Added to a playlist in the playlist page
- Imported if it was exported with an older HxGN Robotic Automation version

A playlist (containing one or more programs) can be :

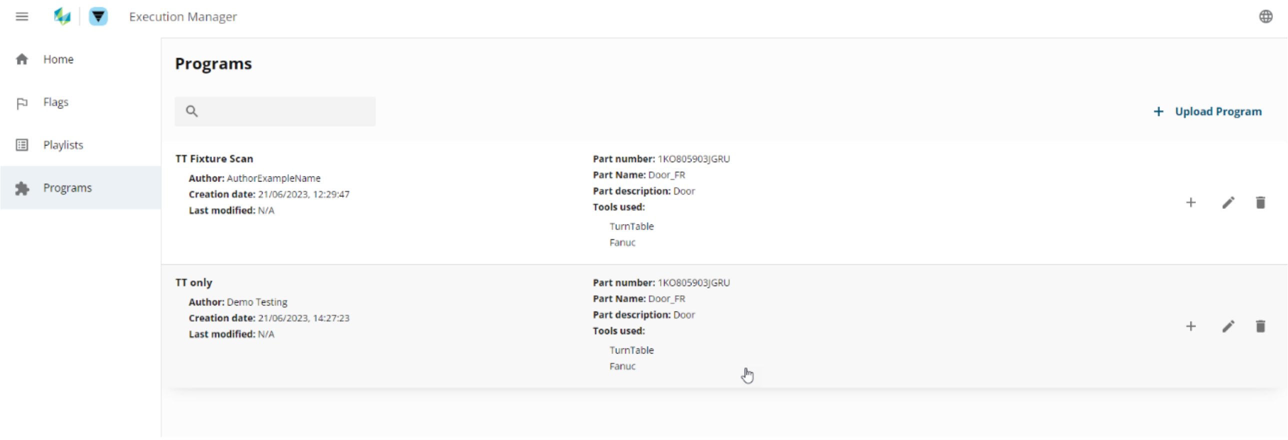
- Added directly to the home page to be executed
- Have flags added in the flags page

Programs and playlists can be executed either in manual or automatic mode:

- In manual mode for both programs and playlist, metadata and the amount of repetitions can be modified by the user after pressing the start button before executing
- In automatic mode:
    - all program and playlist cards are greyed out
    - playlists are executed based on the PLC triggering the flags that have been set
    - metadata is populated from the PLC index that was defined for them

9.2.5 Program page

This page is used to upload old programs and edit programs available in the internal database.



TT Fixture Scan
Author: AuthorExampleName
Creation date: 21/06/2023, 12:29:47
Last modified: N/A

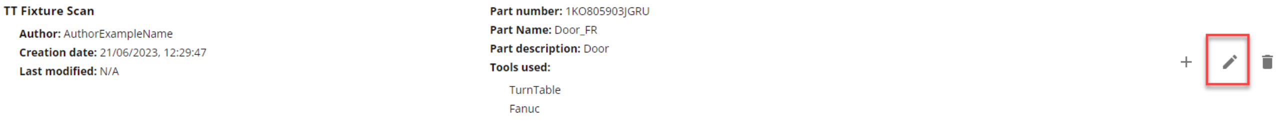
Part number: 1KO805903JGRU
Part Name: Door_FR
Part description: Door
Tools used:
TurnTable
Fanuc

TT only
Author: Demo Testing
Creation date: 21/06/2023, 14:27:23
Last modified: N/A

Part number: 1KO805903JGRU
Part Name: Door_FR
Part description: Door
Tools used:
TurnTable
Fanuc

9.2.5.1 Editing a program

Click on the edit button to open properties page which allows the user to modify the program metadata

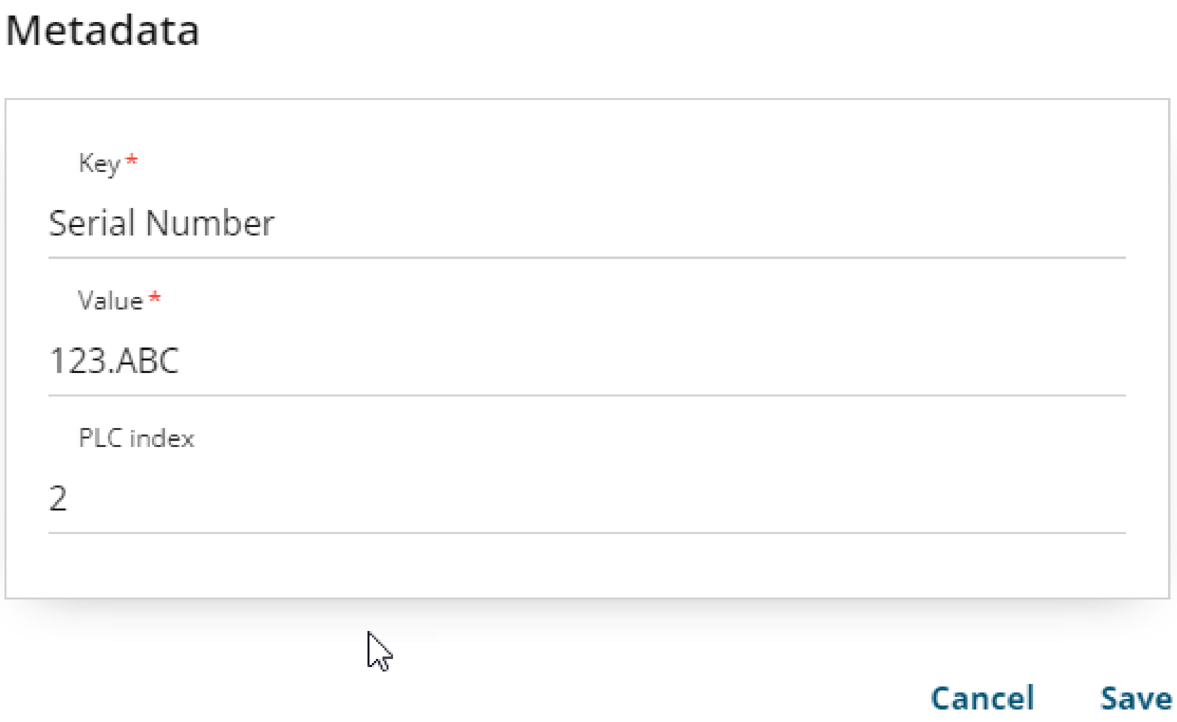


TT Fixture Scan
Author: AuthorExampleName
Creation date: 21/06/2023, 12:29:47
Last modified: N/A

Part number: 1KO805903JGRU
Part Name: Door_FR
Part description: Door
Tools used:
TurnTable
Fanuc

In the program properties the user can edit the Metadata and assign a PLC Index for each field.
If a PLC Index is provided, Execution Manager on execution will search for the field in the PLC at that specified index and copy its contents to the Metadata Value.

In the following example Execution Manger will search in the PLC Index “2” for a value and replace the “123.ABC” value with the value provided by the PLC for the key “Serial Number”.



Metadata

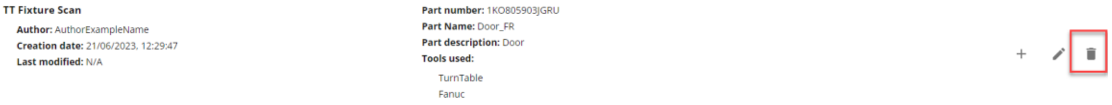
Key *
Serial Number

Value *
123.ABC

PLC index
2

9.2.5.2 Removing a program

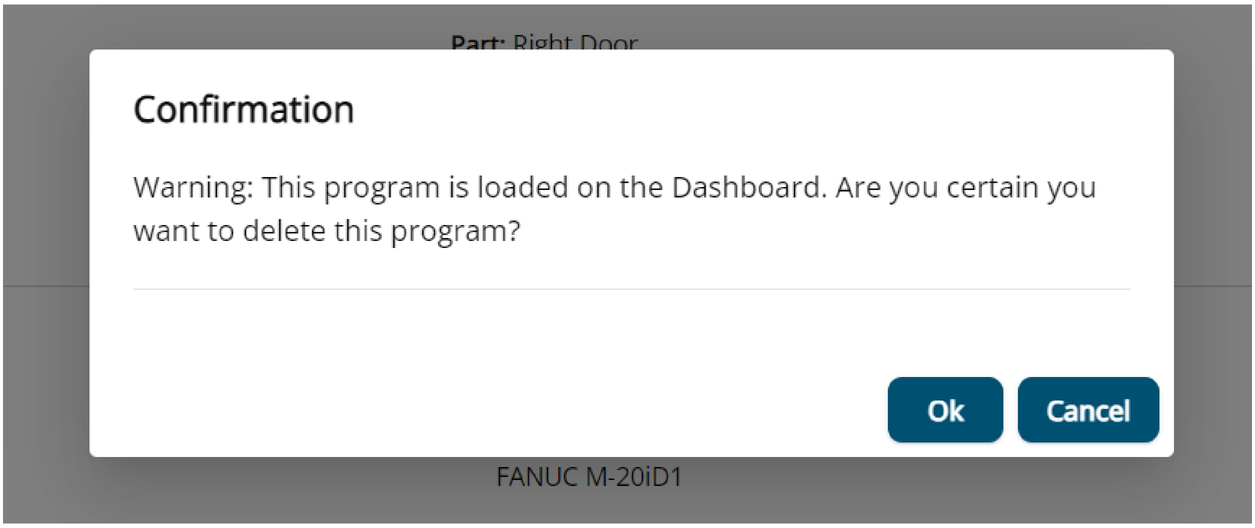
Click on the **Delete** button to remove a program.



TT Fixture Scan
Author: AuthorExampleName
Creation date: 21/06/2023, 12:29:47
Last modified: N/A

Part number: 1KO805903JGRU
Part Name: Door_FR
Part description: Door
Tools used:
TurnTable
Fanuc

If the program is in the home page, but not added to a playlist you will see the following warning message.



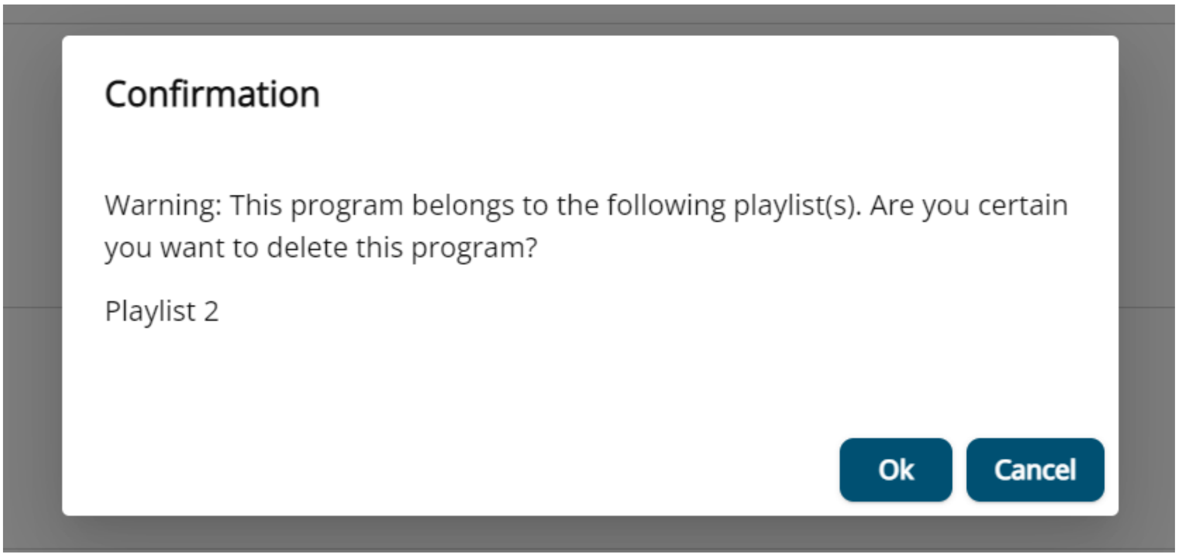
Part: Right Door

Confirmation

Warning: This program is loaded on the Dashboard. Are you certain you want to delete this program?

FANUC M-20iD1

If the program belongs to a playlist you will see the following warning message.

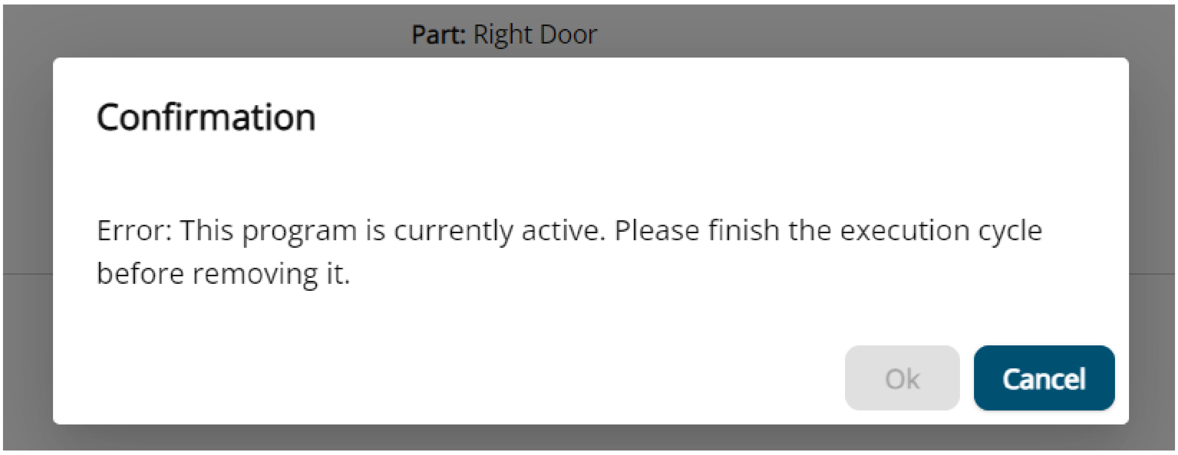


Confirmation

Warning: This program belongs to the following playlist(s). Are you certain you want to delete this program?

Playlist 2

If the program (or playlist containing this program) is running you will see the following warning message.



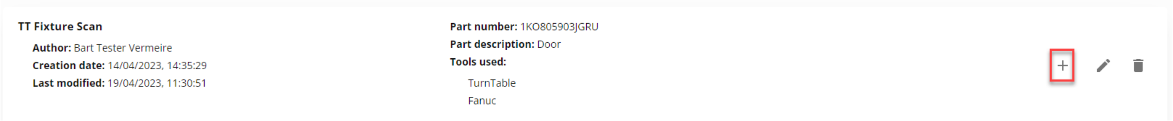
Part: Right Door

Confirmation

Error: This program is currently active. Please finish the execution cycle before removing it.

9.2.5.3 Adding a program directly to the dashboard

Click on the **+** button to add the program directly to the home page.

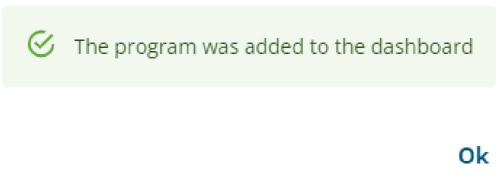


TT Fixture Scan

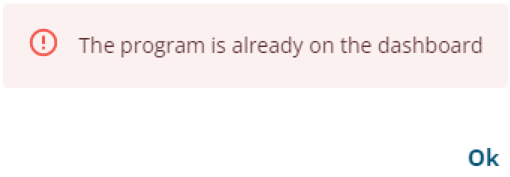
Author: Bart Tester Vermeire
Creation date: 14/04/2023, 14:35:29
Last modified: 19/04/2023, 11:30:51

Part number: 1KO805903JGRU
Part description: Door
Tools used:
TurnTable
Fanuc

A confirmation message will appear or an error in case it already exists on the home dashboard.



The program was added to the dashboard

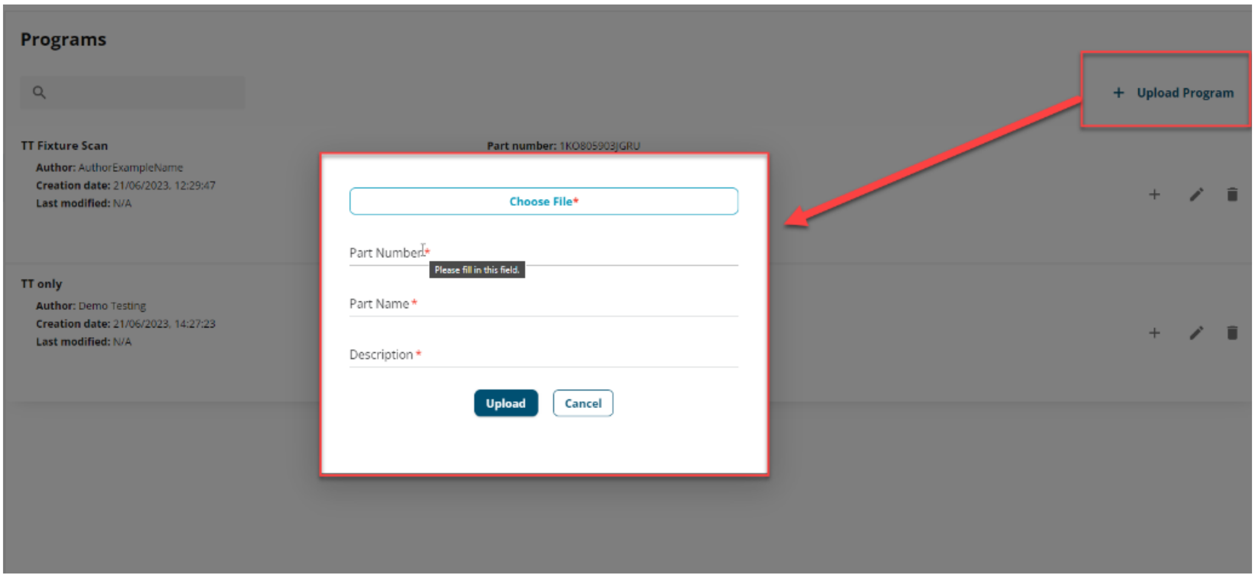


The program is already on the dashboard

9.2.5.4 Importing an already exported *.bfp program

By clicking on the upload program, the user can upload manually a program to the database.
The exported program (*.bfp file) needs to be at least generated by the HxGN Robotic Automation v2.1.2 or higher.
The steps to upload are the following:

1) Click on **Upload Program**
2) Choose the file from your PC
    a) If the Part Number already exist, Execution Manager will auto populate the **Part Name** and **Description** fields
    b) If it is a new Part Number, the user will need to add also the new Part Name and Description



Programs

TT Fixture Scan
Author: AuthorExampleName
Creation date: 21/06/2023, 12:29:47
Last modified: N/A

Part number: 1KO805903JGRU

TT only
Author: Demo Testing
Creation date: 21/06/2023, 14:27:23
Last modified: N/A

Please fill in this field.

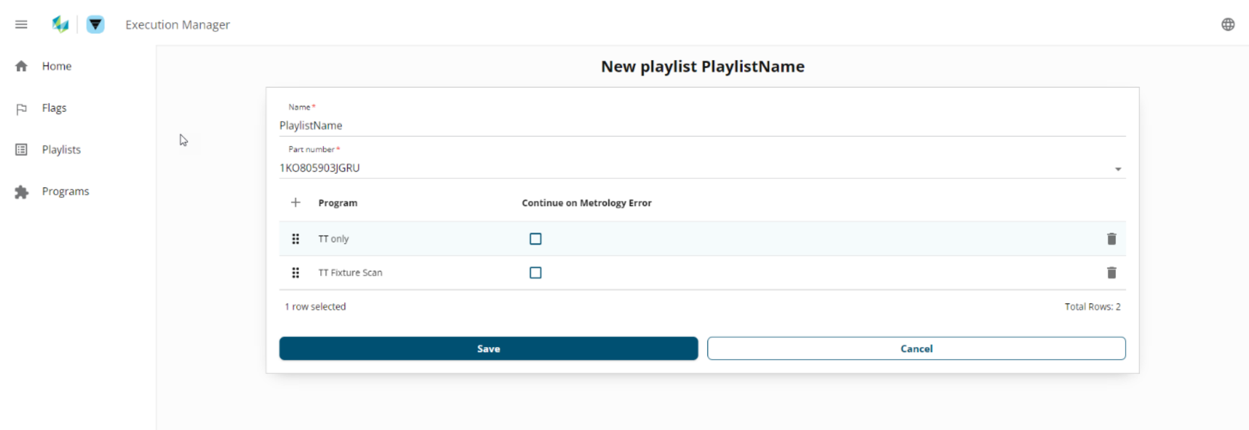
9.2.6 Playlist page

This page is used to create and edit playlists, i.e. sequences of programs to be executed either manually by the operator or automatically triggered by a PLC. How to set a flag to trigger the automatic execution of a playlist is explained in chapter 9.2.7.

9.2.6.1 Creating a playlist

Click on the **Add Playlist** button:

Fill in the new **PlaylistName**, select which **Part number** this playlist is associated to from the drop-down menu. This will show only the programs belonging to that part when the user clicks on the **+** button in the Programs section to add a new program to the playlist.



New playlist PlaylistName

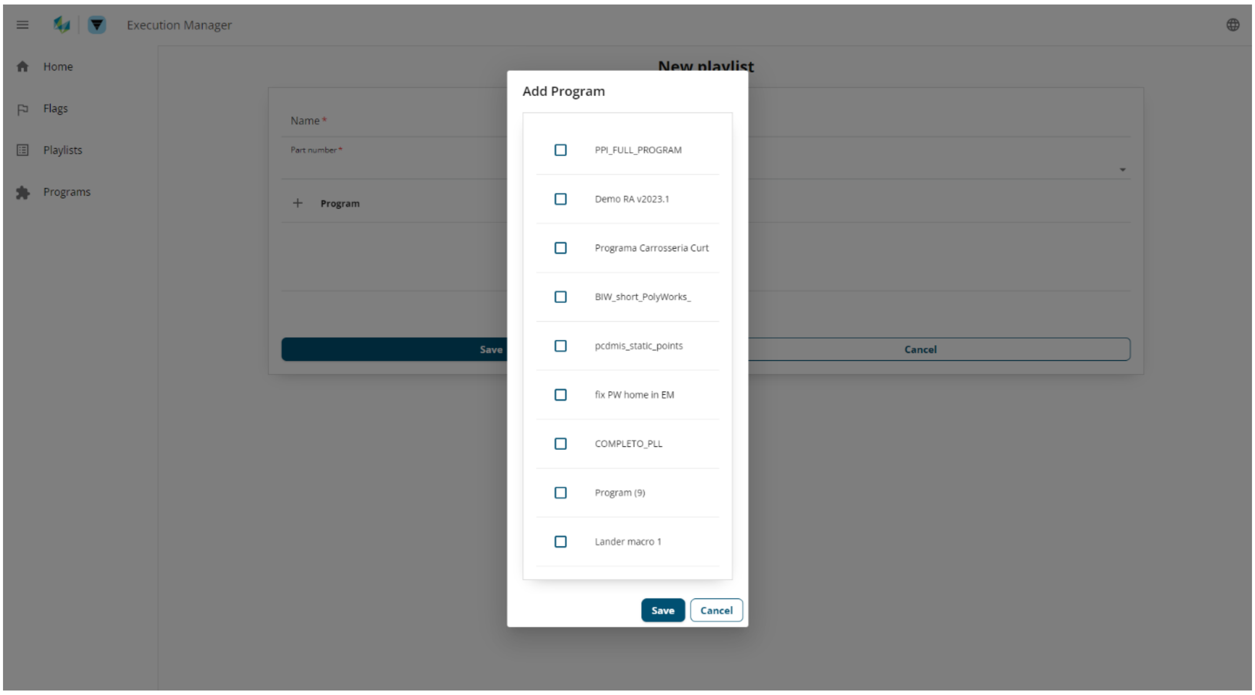
Name *
PlaylistName

Part number *
1KO805903JGRU

	Program	Continue on Metrology Error
	TT only	□
	TT Fixture Scan	□

1 row selected

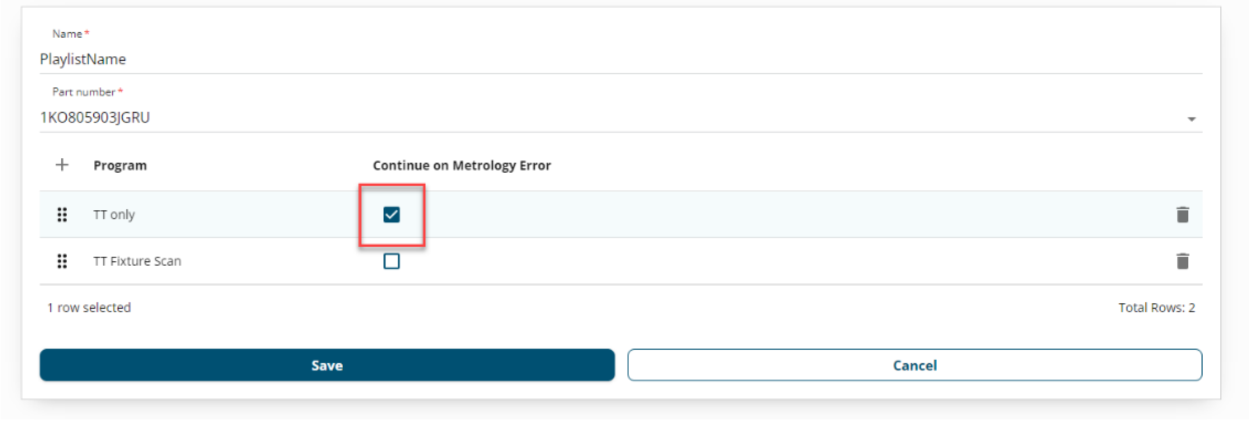
Total Rows: 2



Add Program

□ PPI_FULL_PROGRAM
□ Demo RA v2023.1
□ Programa Carrosseria Curt
□ BIW_short_PolyWorks_
□ pcdmis_static_points
□ fix PW home in EM
□ COMPLETO_PLL
□ Program (9)
□ Lander macro 1

After you added a program to the playlist you can set **Continue On Metrology Error** checkbox for this program. This will cause the playlist either to continue to the next program in the playlist even if there is a Metrology error on the current program or to abort all further programs in the playlist.


Name *
PlaylistName

Part number *
1KO805903JGRU

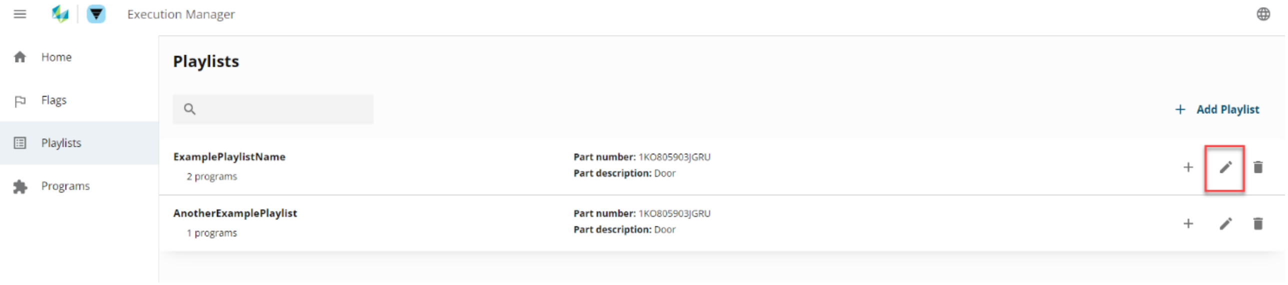
	Program	Continue on Metrology Error
	TT only	☑
	TT Fixture Scan	□

1 row selected

Total Rows: 2

9.2.6.2 Editing a playlist

To edit playlist click the **Edit** button on the playlist. The same playlist editing form as described in the previous section will be shown.



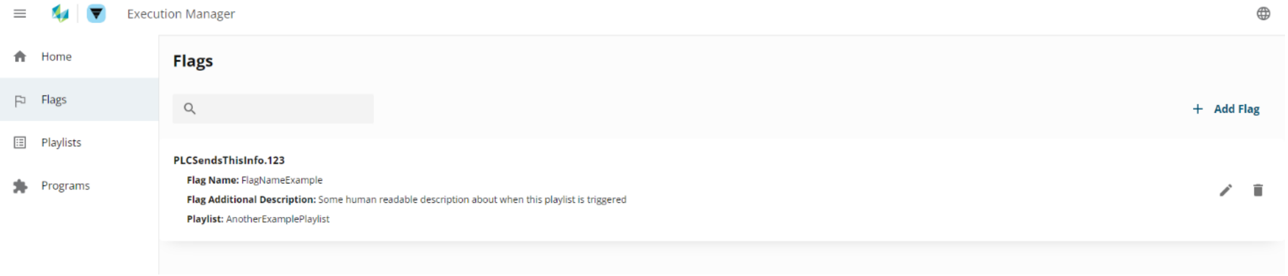
Playlists

ExamplePlaylistName
2 programs
Part number: 1KO805903JGRU
Part description: Door

AnotherExamplePlaylist
1 programs
Part number: 1KO805903JGRU
Part description: Door

9.2.7 Flags page

This page is used to create and edit flags, i.e. PLC values that are associated with a specific playlist execution. When the Execution Manager is set to Automatic mode, it will wait for the PLC to communicate a flag code and execute the associated playlist. Click on **Flags** to navigate to the page showing the Flags registered in the EM:



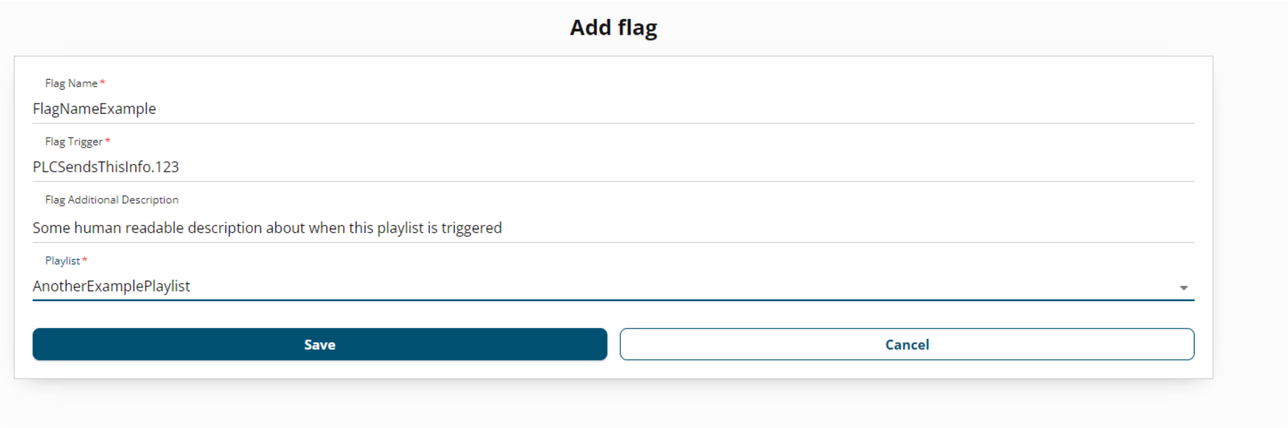
Flags

PLCSendsThisInfo.123
Flag Name: FlagNameExample
Flag Additional Description: Some human readable description about when this playlist is triggered
Playlist: AnotherExamplePlaylist

9.2.7.1 Adding flags

To add new trigger click on the **Add Flag** button:

Fill the **Flag Name**, **Flag Trigger**, **Flag Additional Description** and choose the **Playlist** this Flag must be connected to. The Flag Trigger is what the PLC will send in order to start the corresponding playlist execution.



Add flag

Flag Name *
FlagNameExample

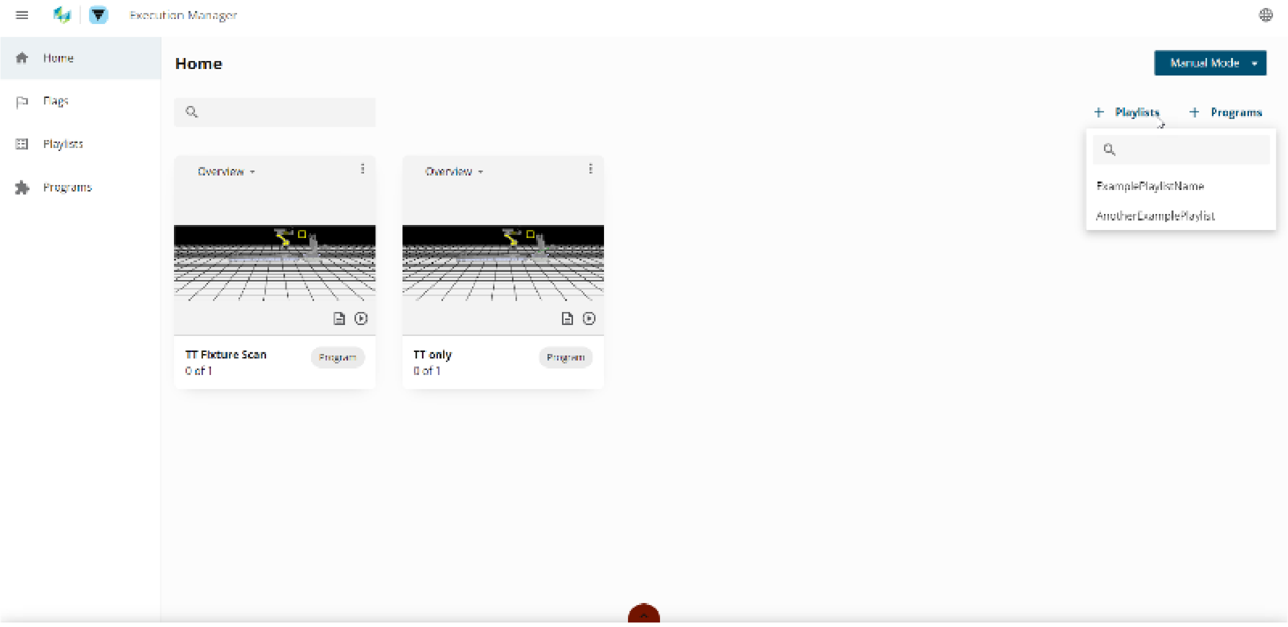
Flag Trigger *
PLCSendsThisInfo.123

Flag Additional Description
Some human readable description about when this playlist is triggered

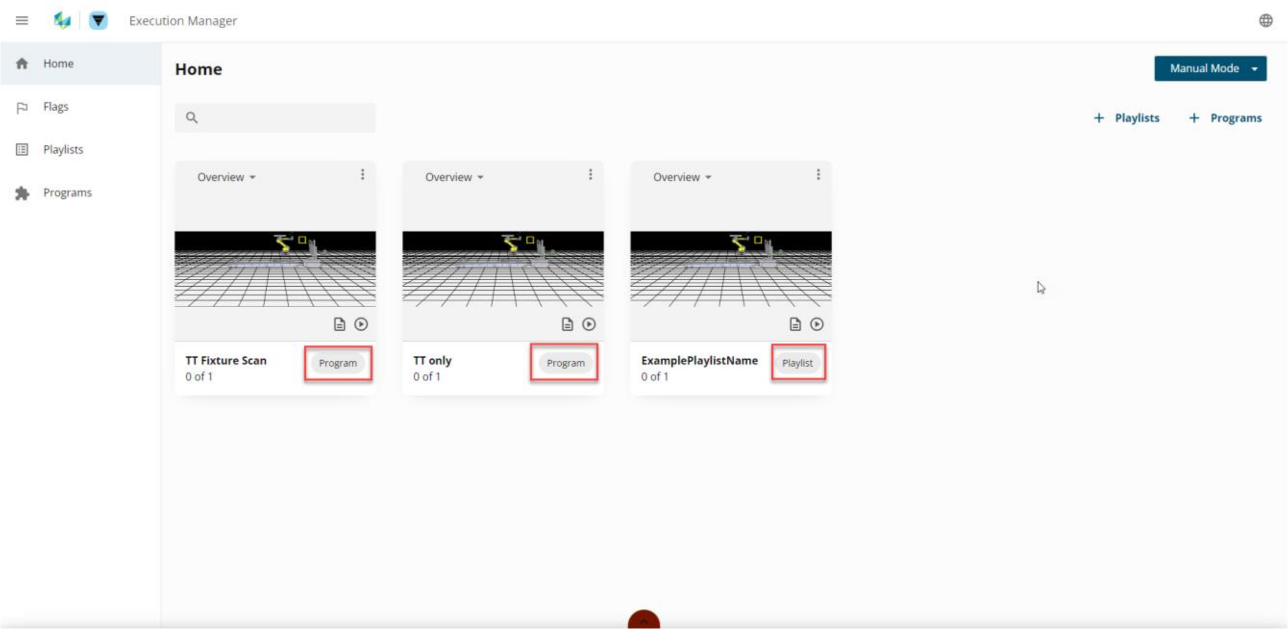
Playlist *
AnotherExamplePlaylist

9.2.8 Home page

In the home page of the application, the user will be able to add either a playlist or a program. There are two separate buttons, one for adding playlists and another for adding programs to the home page. Each of the buttons will open its own selection dialog, with all available playlists or programs. Select **+ Playlists** or **+ Programs** to open a list that can be easily filtered:



The chip within the card allows the user to distinguish between playlists and programs.



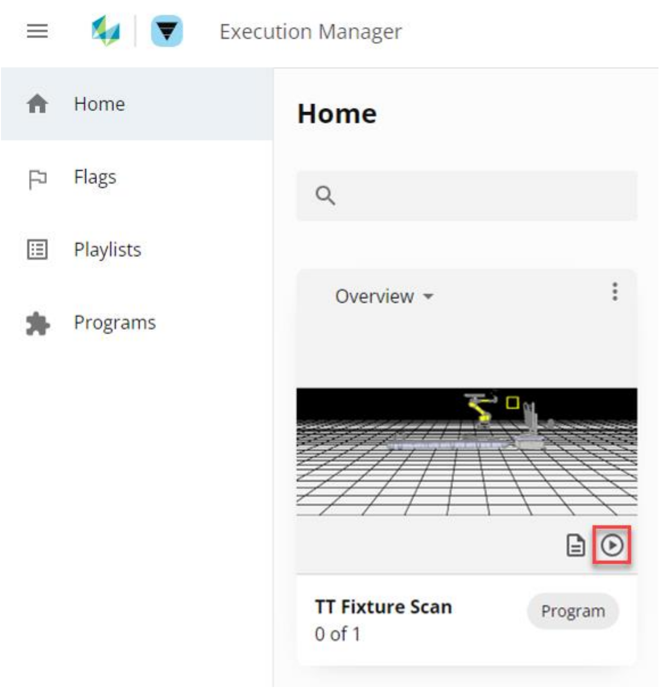
9.2.8.1 Set Automatic or Manual mode

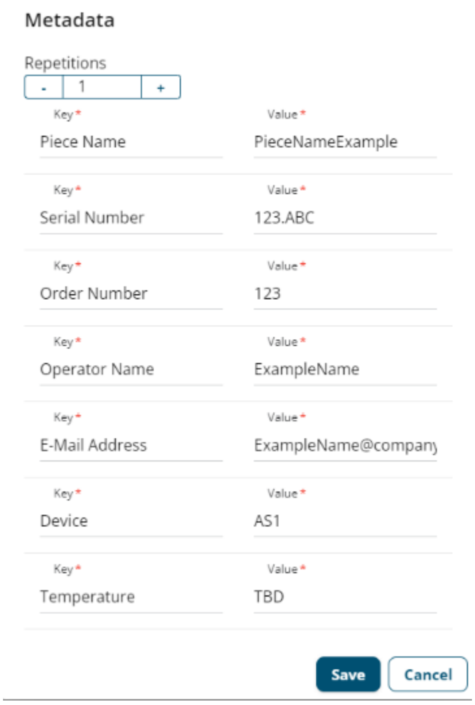
In the home page, on the top right there is an option to select a mode in which a program or a playlist can be executed. There are two options: manual or automatic.

- In automatic mode, the PLC will control the execution of the playlists by sending the relative Flag Code to Execution Manager.
- In manual mode programs and playlists will run in the order defined by the user.

9.2.8.2 Executing a program or a playlist in manual mode

In manual mode, after the user presses on the **Execute** button (▶), a new form will appear allowing to set the number of repetitions for the program. In manual mode the form will also contain metadata that the user can edit before submitting the execution.



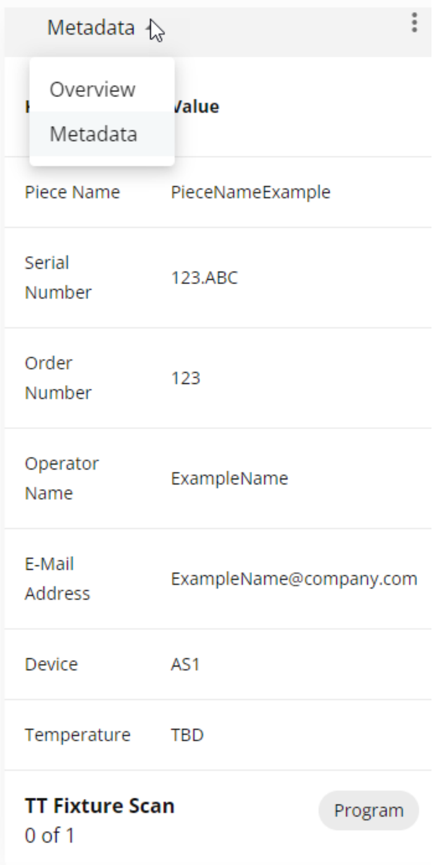


Metadata

Repetitions
- 1 +

Key*	Value*
Piece Name	PieceNameExample
Serial Number	123.ABC
Order Number	123
Operator Name	ExampleName
E-Mail Address	ExampleName@company
Device	AS1
Temperature	TBD

Additionally, on program and playlist cards there is a drop-down menu, on which the user can choose to visualize the metadata of the selected program or playlist.



	Value
Piece Name	PieceNameExample
Serial Number	123.ABC
Order Number	123
Operator Name	ExampleName
E-Mail Address	ExampleName@company.com
Device	AS1
Temperature	TBD

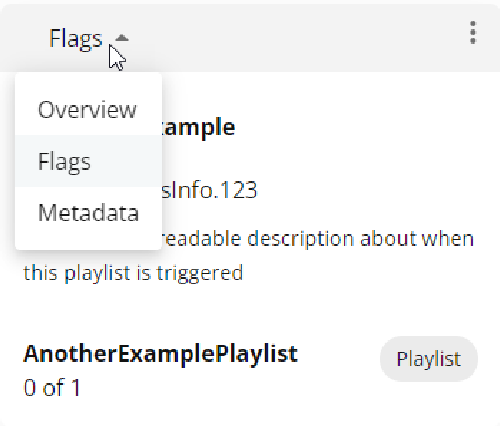
9.2.8.3 Executing a playlist in automatic mode

All the programs or playlists with no triggers will be greyed out and cannot be selected. The playlists with triggers are the ones Execution Manager will be monitoring.

Once a trigger is sent by the PLC, Execution Manager will load the playlist and confirm the part with the PLC before starting the program using the Metadata populated by the PLC, fetching it depending on the flag code.

The PLC needs to send the flag code in order to run the playlist associated with it. That is why in automatic mode programs and playlists without flag codes are greyed out, to let the user know that those programs or playlists cannot run automatically.

The flags can be visualized by using the drop-down menu in the Card view and selecting the **Flags** option:



:ample

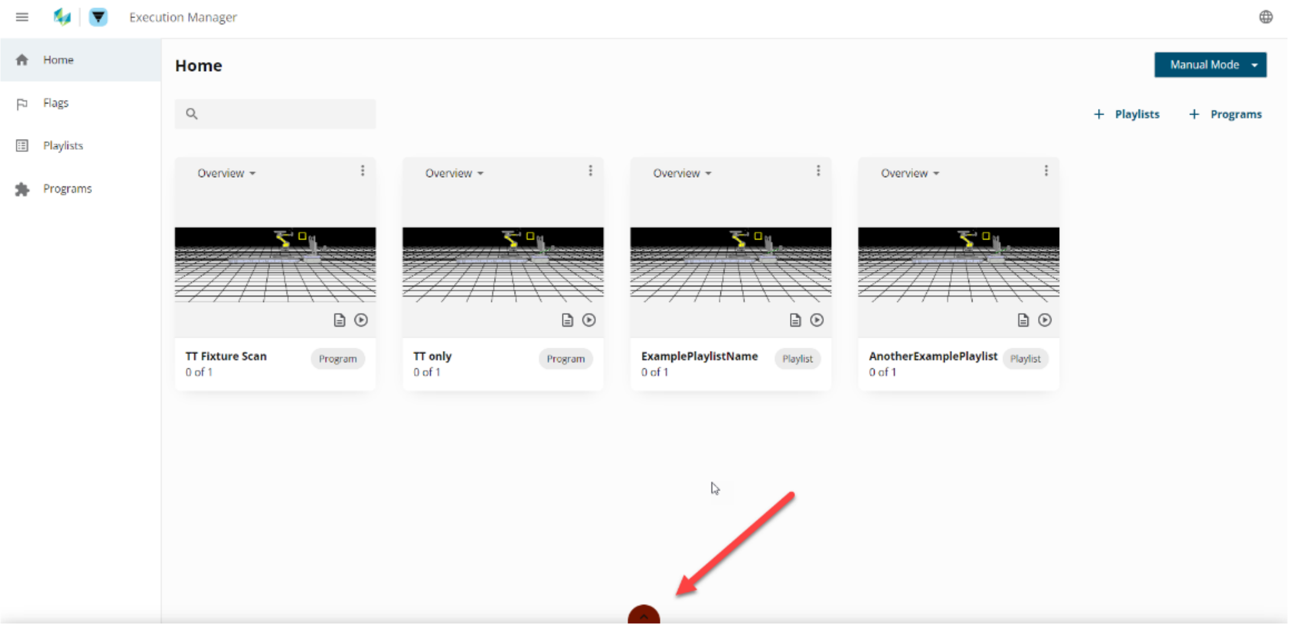
sInfo.123

readable description about when this playlist is triggered

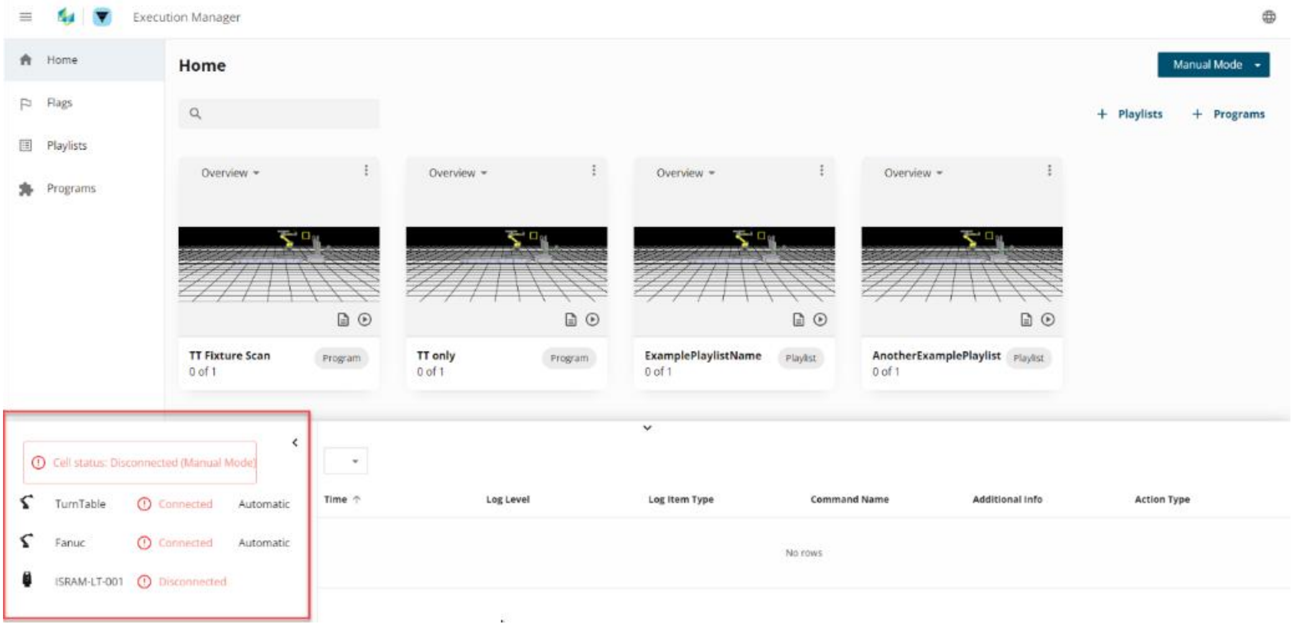
AnotherExamplePlaylist
0 of 1

9.2.9 Cell status and program logs

The cell status will show red in case not all the cell components are connected are running in automatic mode, meaning that the Execution Manager is allowed to take control of the cell and execute programs or playlists.



By expanding the flyout, the user can view the cell status.

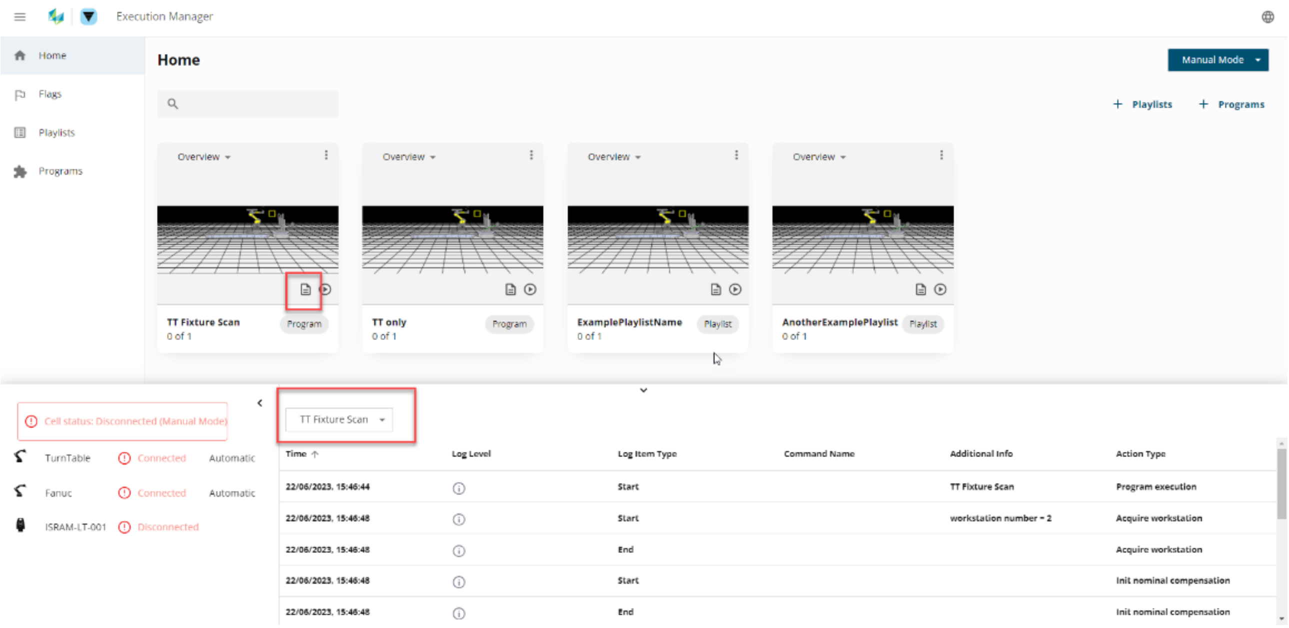


Cell status: Disconnected (Manual Mode)

TurnTable	Connected	Automatic
Fanuc	Connected	Automatic
ISRAM-LT-001	Disconnected	

Time ↑	Log Level	Log Item Type	Command Name	Additional Info	Action Type
No rows					

By expanding the cell status window, Execution Manager also shows the program logs of the programs that have been executed.
Alternatively the log drop-down menu allows to filter the logs depending on the related program/playlist, showing the history of their last execution.



Cell status: Disconnected (Manual Mode)

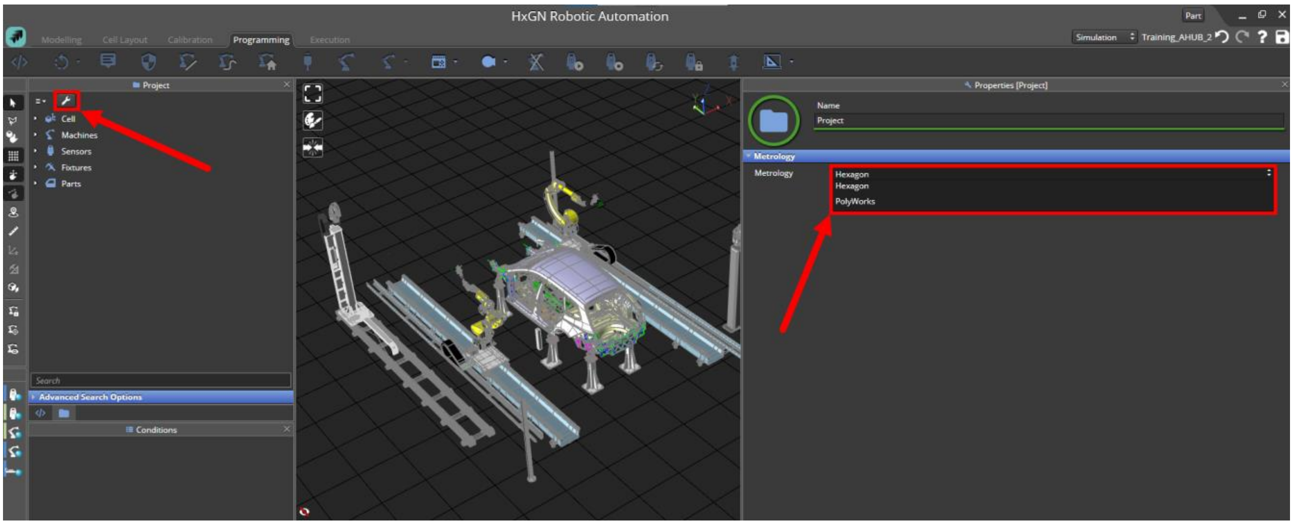
TurnTable	Connected	Automatic
Fanuc	Connected	Automatic
ISRAM-LT-001	Disconnected	

TT Fixture Scan

Time ↑	Log Level	Log Item Type	Command Name	Additional Info	Action Type
22/06/2023, 15:46:44	ⓘ	Start		TT Fixture Scan	Program execution
22/06/2023, 15:46:48	ⓘ	Start		workstation number = 2	Acquire workstation
22/06/2023, 15:46:48	ⓘ	End			Acquire workstation
22/06/2023, 15:46:48	ⓘ	Start			Init nominal compensation
22/06/2023, 15:46:48	ⓘ	End			Init nominal compensation

10 Metrology

It is possible to define the metrology application used in the HxGN Robotic Automation project by selecting it using the relative **Metrology** drop-down menu. Currently, HxGN Robotic Automation supports PC-DMIS, Inspire and PolyWorks as metrology applications. Clicking on **Project Properties** in HxGN Robotic Automation, user is able to select either Hexagon (if using PC-DMIS or Inspire) or PolyWorks.



Name: Project

Metrology: Hexagon

10.1 PC-DMIS

HxGN Robotic Automation 2023.1 is compatible with the following PC-DMIS 2023.1 executables:

PC-DMIS must be licensed with the following minimum required options:

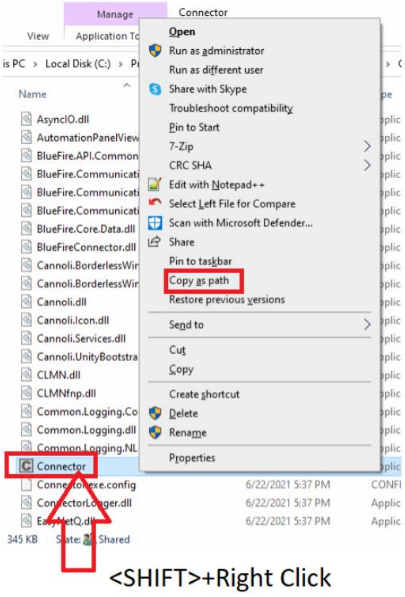
- ROY AUTOMATE
- CAD++ OFFLINE
- CMS106 Laser Interface

Recommended options depending on the application or CAD package in use by specific customer are:

- CSV-Converter
- Toolkit (ESF)
- SolidWorks
- Catia V5 or V6

Configure the PCDMIS metrology software with the following steps:

1) Open File Explorer and look for the following file: “C:\ProgramData\Hexagon\RA\Config\RAconfig.xml”
2) Edit the file “RAConfig.xml” with notepad editor with Admin rights, making a backup first.
3) Look for line containing: “ConfigSetting Name="MetrologyAppPath” …
4) Copy the path where the Connector.exe is ("C:\Program Files\Hexagon\PCDMIS 2023.1 64-bit\Connector\Connector.exe") – [Shift] + right click on Connector.exe and then click on “Copy as path”.

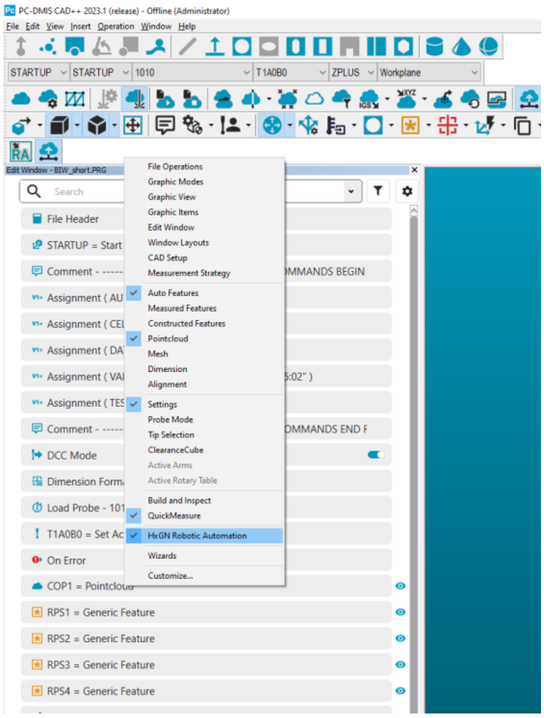


5) Paste the copied path inside the “Value” tag of “ConfigSetting Name="MetrologyAppPath” field inside “RAConfig.xml” file and save the changes.

10.1.1 Workflow using PC-DMIS

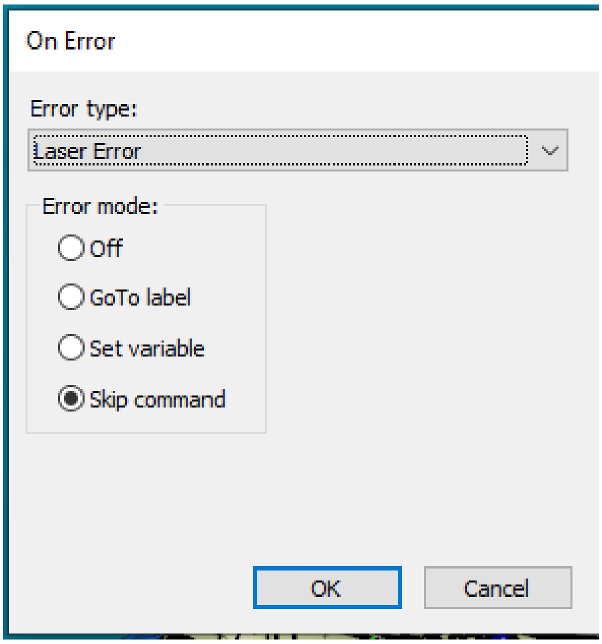
The cell layout must be ready, aligned and the robot calibrated in the project. For more information about how to align the cell and calibrate the robot refer to chapter 7.

The PC-DMIS program must be ready and prepared with the features needed for the inspection plan. To see the HxGN Robotic Automation toolbar, go to *View* | *Toolbars* | *HxGN Robotic Automation*.



In the PC-DMIS program do the following steps:

1) Open PC-DMIS application.
2) Type **ONERROR/Laser_error** in the program to skip all the error dialog boxes popping up in case of errors.
3) Optionally set **SKIP** in the program configuration to obtain a list of the features that have not been measured, after the program is executed.



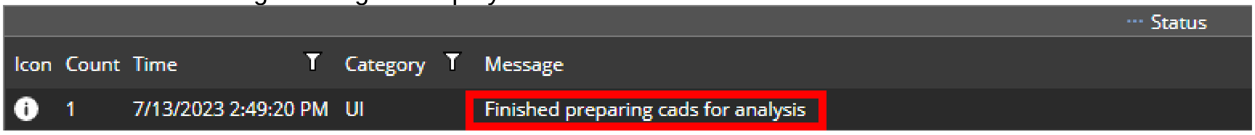
On Error

Error type: Laser Error

Error mode:

○ Off
○ GoTo label
○ Set variable
● Skip command

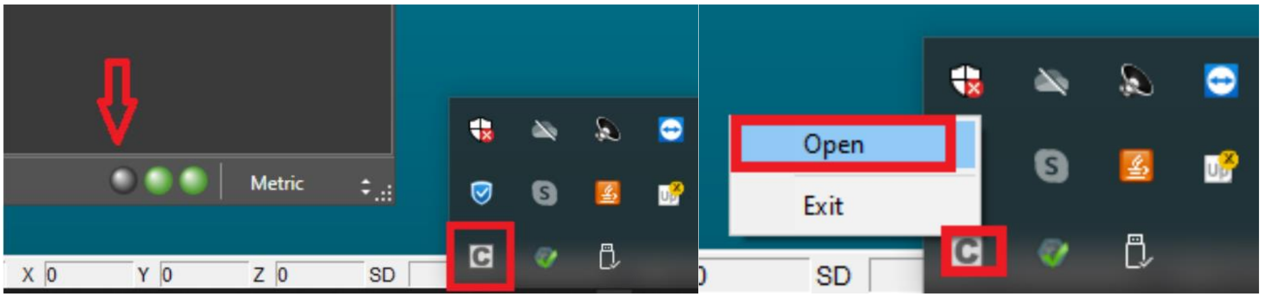
4) Create an empty COP.
5) Create a PC-DMIS measurement routine using a laser probe and make sure you have the proper PC-DMIS license. You can either manually create the measurement routine or import a CSV version with the CSV-Converter application.
6) Open HxGN Robotic Automation application.
7) Wait until the following message is displayed within the Status window.



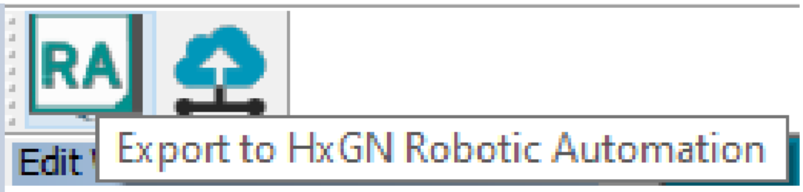
Status

Icon	Count	Time	Category	Message
ⓘ	1	7/13/2023 2:49:20 PM	UI	Finished preparing cads for analysis

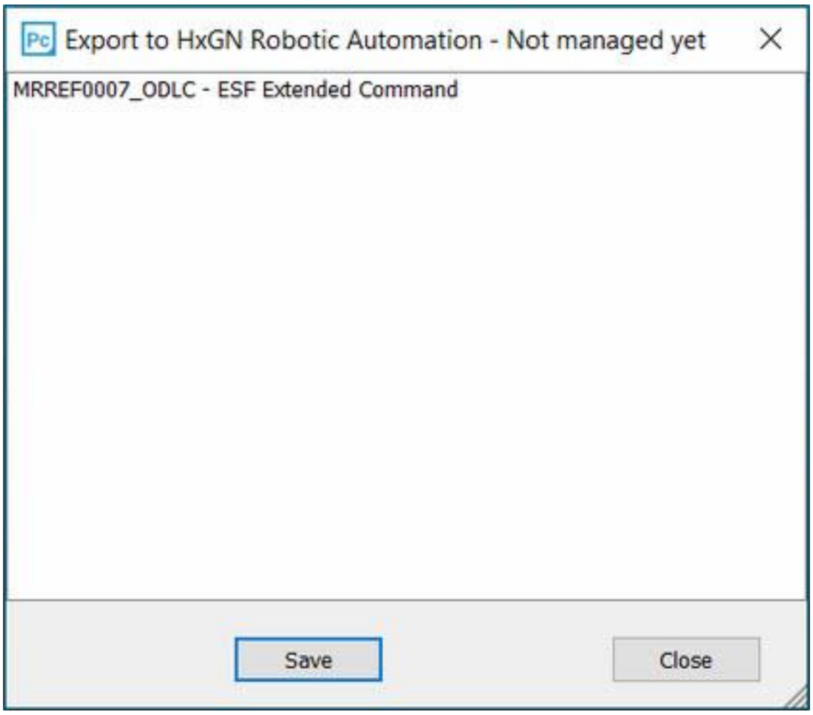
8) In HxGN Robotic Automation click on the left traffic light related to Metrology Software to open the Connector. The icon is grey if there is no connection established.
9) After clicking it, the colour will change to yellow during the connection process and then turn to green when the connection is established.
10) If there is any error during the connection, the icon will turn red. It is possible to open the Connector window to check the status by clicking on the relative icon within the hidden icons in the taskbar. See PC-DMIS documentation for more details on how to configure the connector application.



11) In PC-DMIS within the HxGN Robotic Automation toolbar, click the **Export to HxGN Robotic Automation** button or select the *File |Export | HxGN Robotic Automation* menu option.



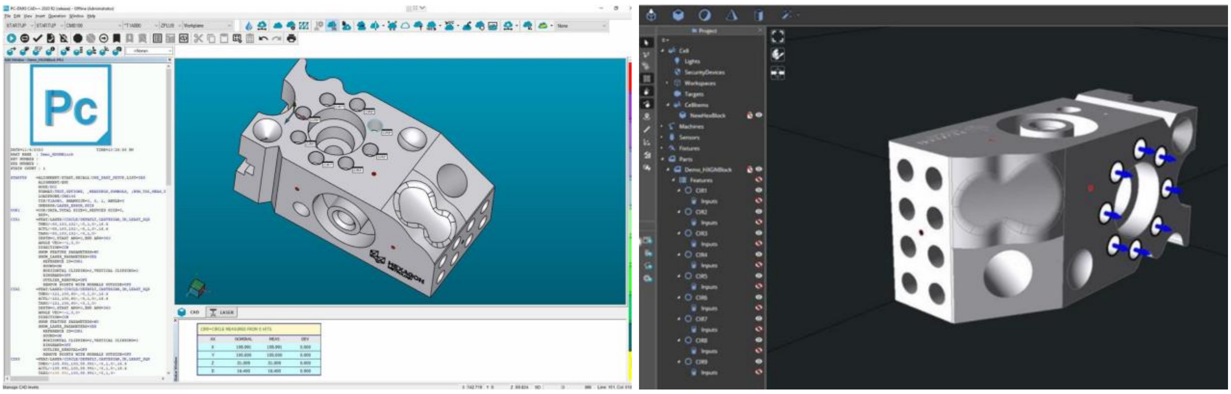
For features not yet supported, HxGN Robotic Automation displays the “Export to HxGN Robotic Automation - Not managed yet” dialog box to list them as shown in the following picture:



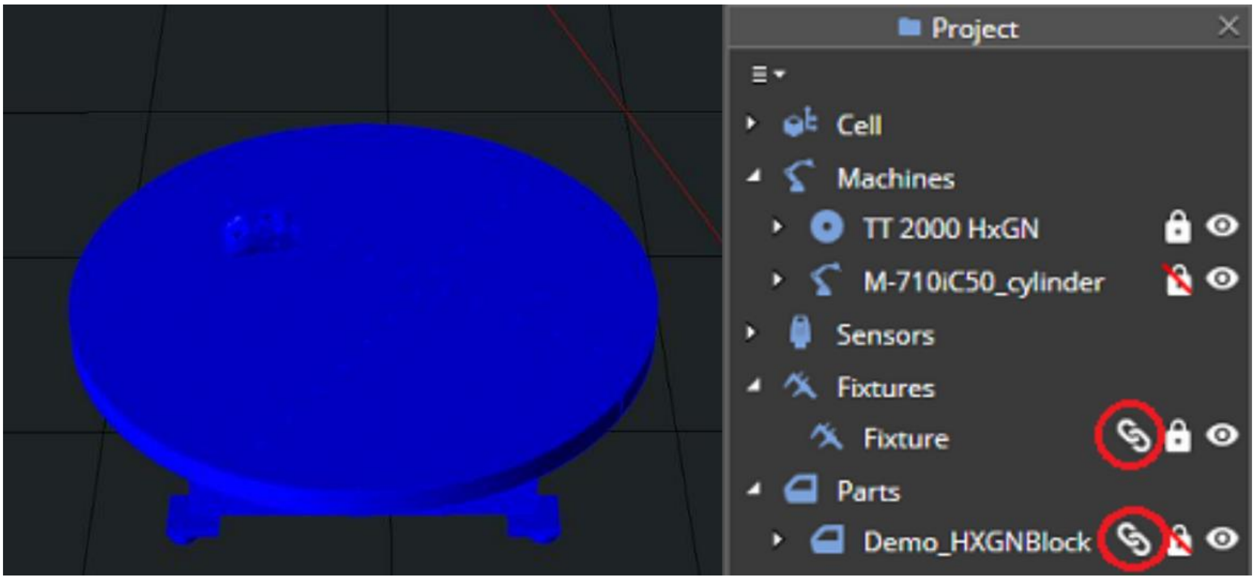
MRREF0007_ODLC - ESF Extended Command

Click **Save** to save the report of unsupported features as a text file.

12) The part and the features will be sent to HxGN Robotic Automation application.



13) Align the part in your cell in HxGN Robotic Automation, e.g., if you are using a Turn Table, assign the part to the fixture and the fixture to the Turn Table.



14) Once the program is ready to execute, from the HxGN Robotic Automation toolbar in PC-DMIS, click the **TCP/IP Pointcloud Server receive data button** () to open the communication channel between PC-DMIS and HxGN Robotic Automation and send the COP from HxGN Robotic Automation to PC-DMIS.
15) From HxGN Robotic Automation, create and execute the scanning program. PC-DMIS receives the scanned data from the HxGN Robotic Automation, and automatically executes the measurement routine. You can send scanned data to PC-DMIS to obtain measurement results and perform data analysis and generate metrology reports.

10.1.2 PC-DMIS Static point measurement

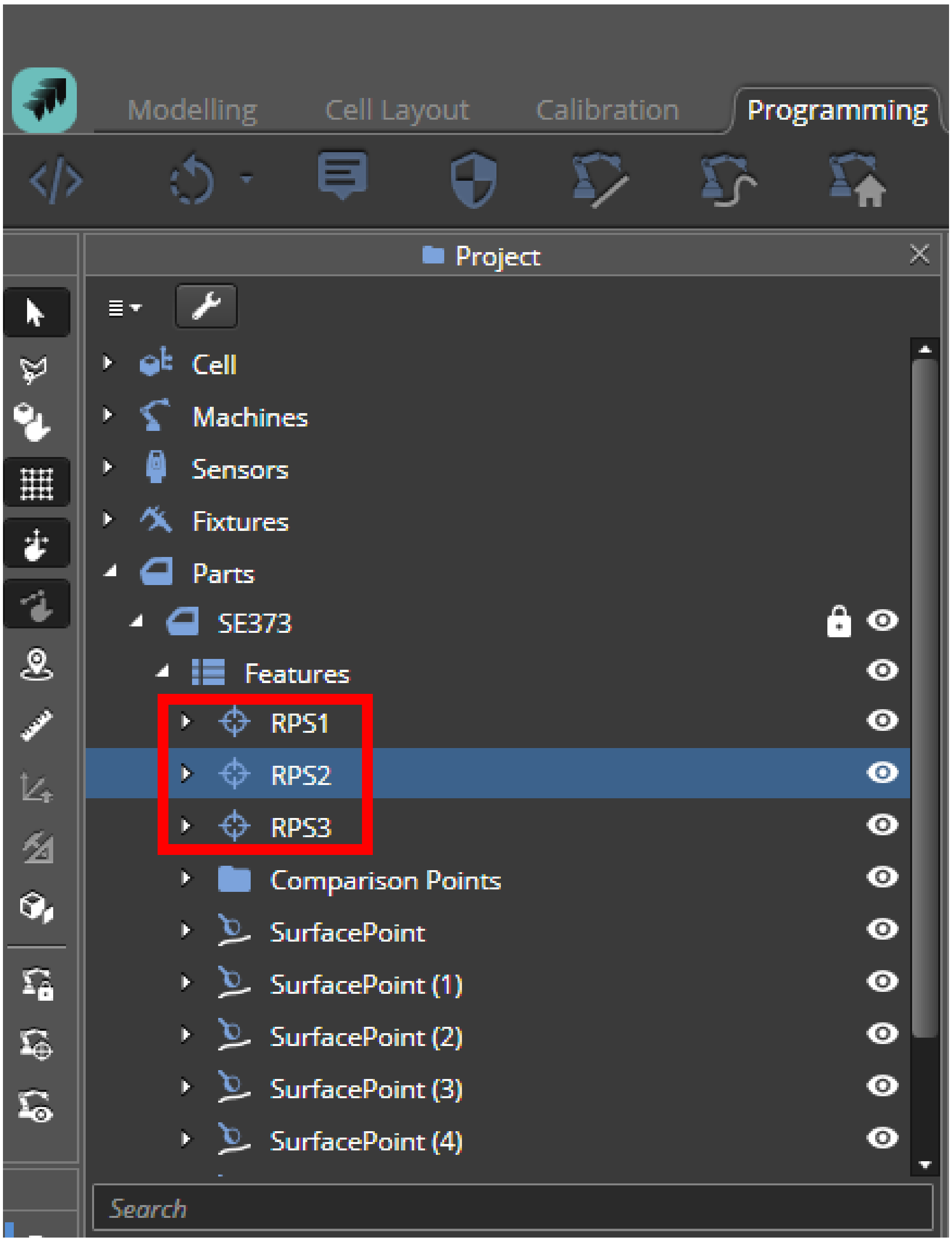
For alignment, Metrology applications use a RPS (Reference Point System) defined usually by specific alignment features (e.g. for a BIW by two circles and a slot) to align the part with respect to the RPS. However for a BIW these alignment features are usually under the vehicle and not easy to scan.

To solve this situation, in PC-DMIS metrology application the user can define some generic static point features and these can be linked in HxGN Robotic Automation to reflectors.

The user can then create a program to measure these reflector positions and send the measurement to PC-DMIS that will be able to use them for alignment.

Below it is described the workflow to use static points for alignment using PC-DMIS metrology application.

1) Open the HxGN Robotic Automation application and connect to the PC-DMIS metrology software.
2) Using PC-DMIS define the static points as generic features that will be used for alignment and will be sent to HxGN Robotic Automation as special features defined by the ⌖ icon.

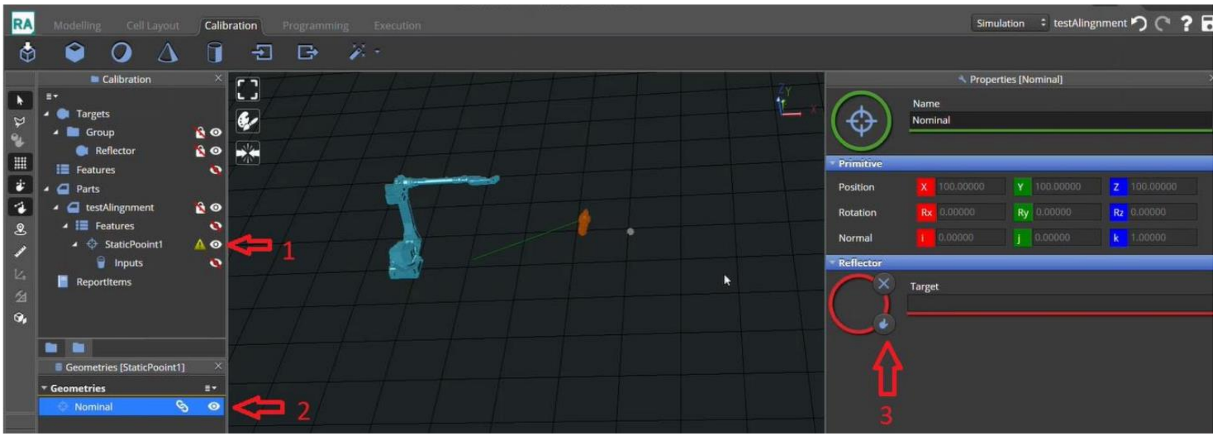


3) In HxGN Robotic Automation define a reflector using the **Add Reflector Wizard** (see chapter 7.2.5).

Warning: In HxGN Robotic Automation the newly added static points must be linked to a reflector. If a static point is not linked to a reflector, then a warning will appear beside the feature and validation will fail.

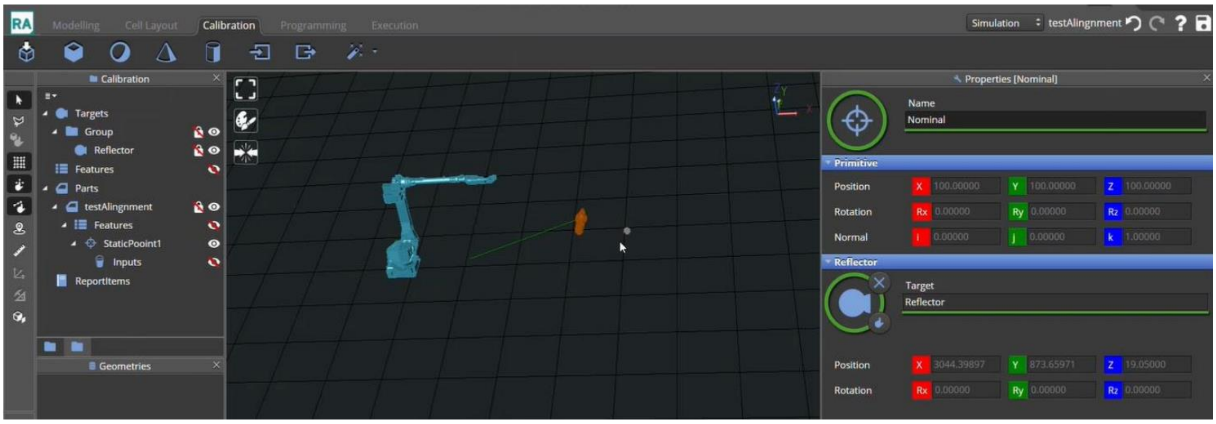
Info: The static point feature is linked automatically to the reflector which is in the exact same coordinates as the feature.

4) If there is no reflector in the static point position, there are four options:
   a) Change the static point position through PC-DMIS to match the reflector position.
   b) Change the reflector position in HxGN Robotic Automation to match the static point position defined in PC-DMIS.
   c) Link the feature manually to a reflector by drag and drop in the project tree.



Name: Nominal

Position	X 100.00000	Y 100.00000	Z 100.00000
Rotation	Rx 0.00000	Ry 0.00000	Rz 0.00000
Normal	i 0.00000	j 0.00000	k 1.00000



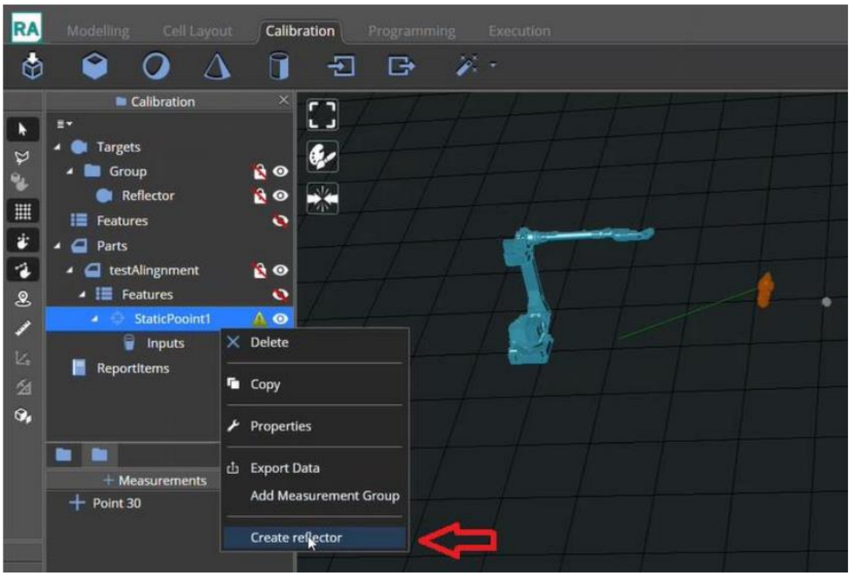
Name: Nominal

Position	X 100.00000	Y 100.00000	Z 100.00000
Rotation	Rx 0.00000	Ry 0.00000	Rz 0.00000
Normal	i 0.00000	j 0.00000	k 1.00000

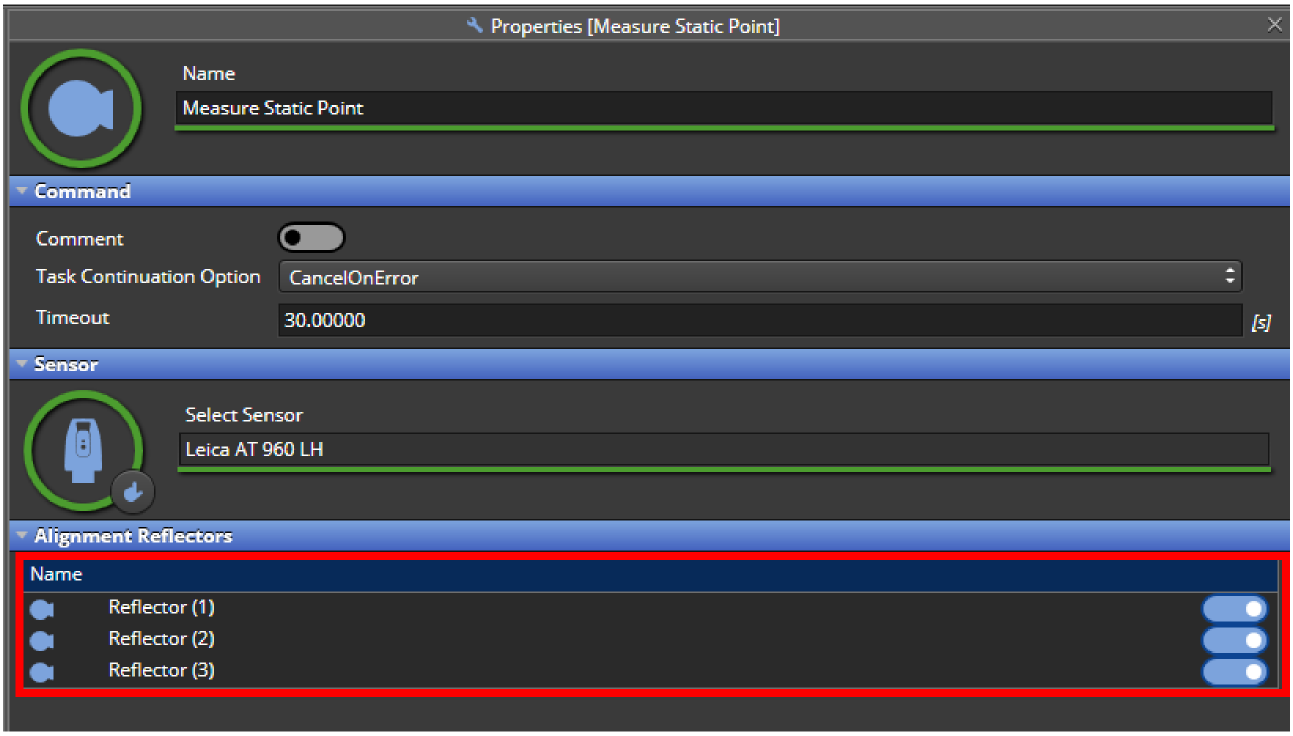
Target: Reflector

Position	X 3044.39897	Y 873.65971	Z 19.05000
Rotation	Rx 0.00000	Ry 0.00000	Rz 0.00000

   d) Create a reflector using the feature position by right clicking on it and select **Create reflector**:



The **Measure PC-DMIS Static Points** command must be used to measure the static point positions, specifying in the properties which reflectors must be used. For the sake of accuracy, it is preferable to measure all the reflectors used for alignment with a single command.



Properties [Measure Static Point]

Name: Measure Static Point

Command

Comment: ○
Task Continuation Option: CancelOnError
Timeout: 30.00000 [s]

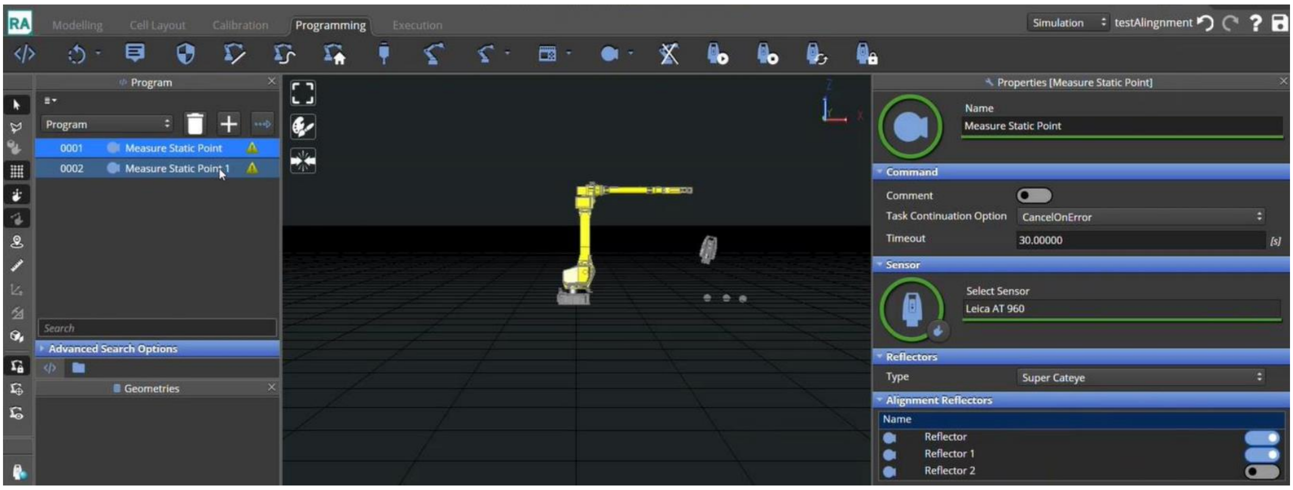
Sensor

Select Sensor: Leica AT 960 LH

Alignment Reflectors

Name	
Reflector (1)	●
Reflector (2)	●
Reflector (3)	●

During the validation of a program having **Measure PC-DMIS Static Points** commands, HxGN Robotic Automation will check if the command covers all the reflectors used for alignment, e.g., three of these reflectors can be covered with one command that measures all of them, or by two commands: the first one measuring two reflectors and the second one measuring the remaining reflector.



Program

0001	Measure Static Point	⚠
0002	Measure Static Point 1	⚠

Properties [Measure Static Point]

Name: Measure Static Point

Command

Comment: ○
Task Continuation Option: CancelOnError
Timeout: 30.00000 [s]

Sensor

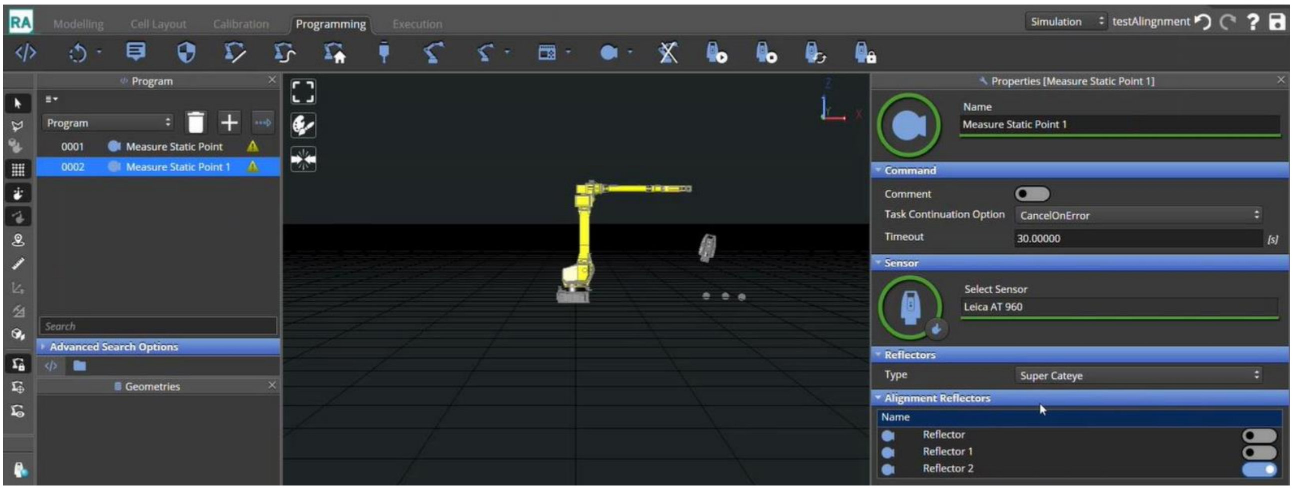
Select Sensor: Leica AT 960

Reflectors

Type: Super Cateye

Alignment Reflectors

Name	
Reflector	●
Reflector 1	●
Reflector 2	○



Program

0001 Measure Static Point

0002 Measure Static Point 1

Name: Measure Static Point 1

Comment

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]

Select Sensor: Leica AT 960

Type: Super Cateye

Name
Reflector
Reflector 1
Reflector 2

If less than three reflectors are measured validation will show a warning.
If more than three reflectors are measured (this will happen if the same reflector is measured multiple times) validation will also show a warning.
In case there are no static points the command **Measure PC-DMIS Static Points** cannot be validated.
In case the state of the static points is changed (removed/added), validation will reset and the program will need to be validated again.
When executing the program, the measured reflectors are sent to PC-DMIS for alignment calculation. In case of an issue, PC-DMIS can stop the program execution.

10.2 Inspire

HxGN Robotic Automation 2023.1 is compatible with Inspire 2023.1.
Inspire inspection software is a single interface, which supports a wide range of portable probing and scanning devices.
For more details, please refer to:
https://www.hexagonmi.com/products/metrology-software/inspire
https://discoverinspire.com/download/InspireReferenceGuide.pdf

HxGN Robotic Automation application can support connection to Inspire Metrology software to perform Measurement Analysis.
Obtain a license and the correct official software version release which supports that connection.

Configure the Inspire metrology software with the following steps:

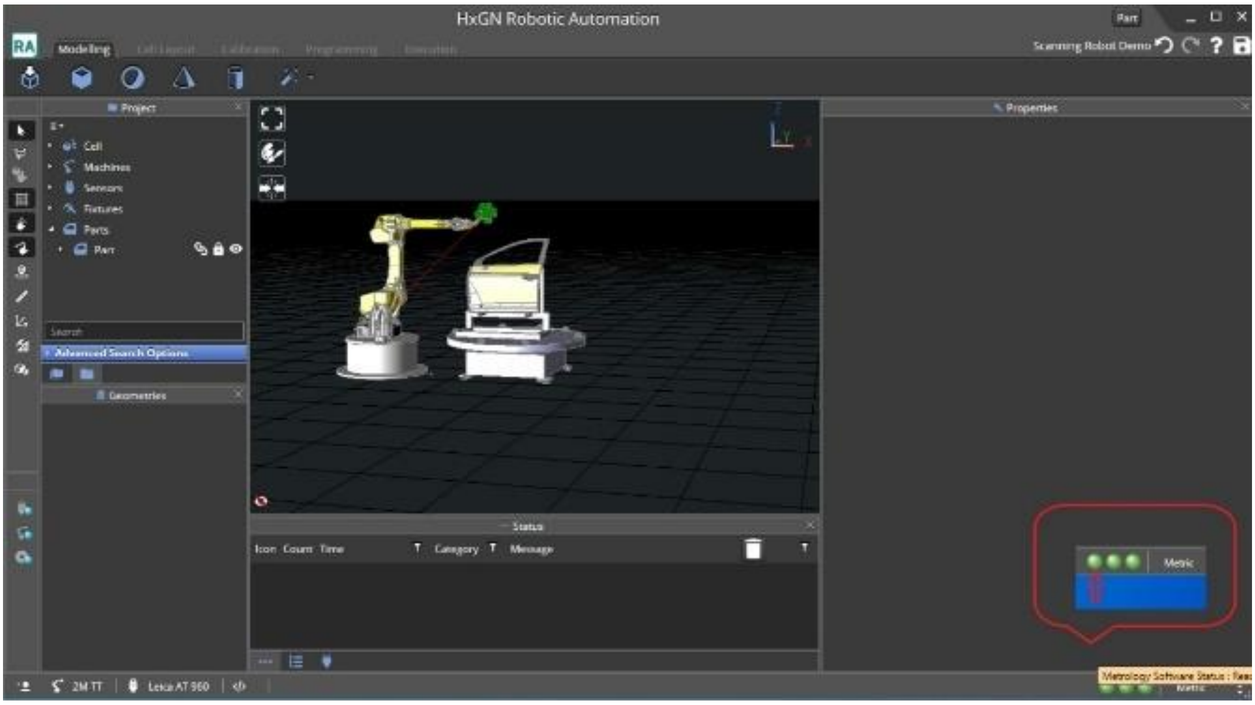
1) Open File Explorer and look for the following file: “C:\ProgramData\Hexagon\RA\Config\RAconfig.xml”
2) Edit the file “RAConfig.xml” with notepad editor with Admin rights, making a backup first.
3) Look for line containing: “ConfigSetting Name="MetrologyAppPath” …
4) Copy the path where the Inspire.exe is (C:\Program Files\Inspire\Inspire (2023.1.161.0)\Inspire.exe”)
5) Paste the copied path inside the “Value” tag of “ConfigSetting Name="MetrologyAppPath” field inside “RAConfig.xml” file and save the changes.

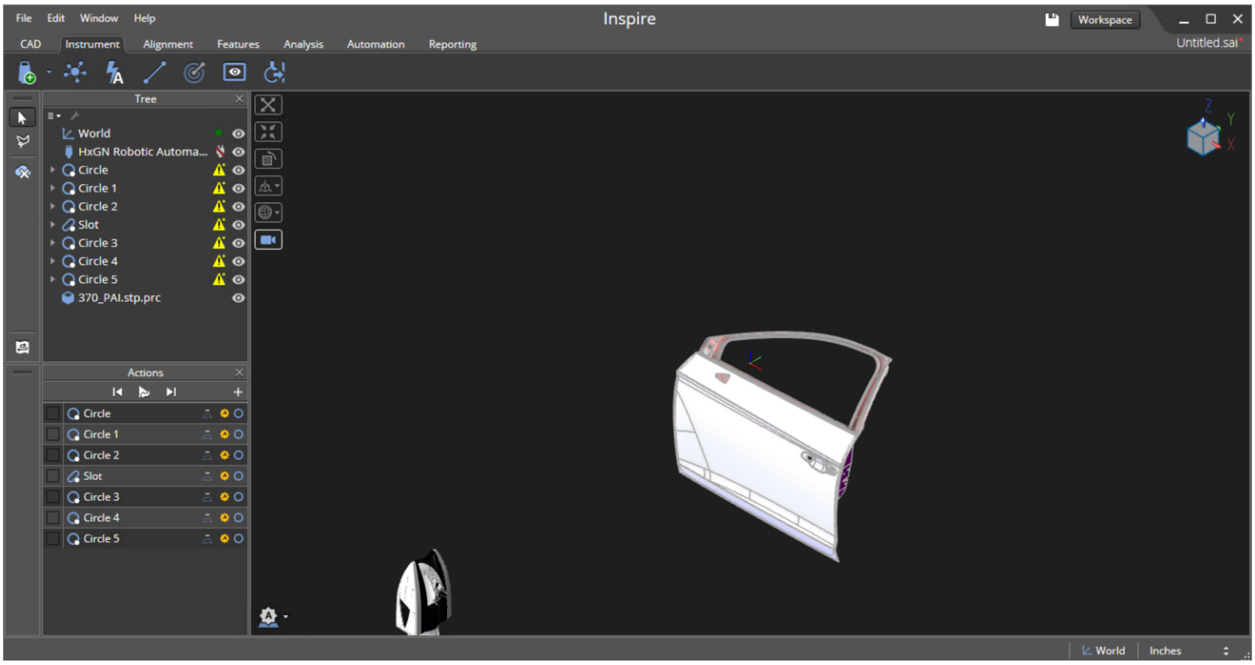
10.2.1 Workflow using Inspire

1) Open HxGN Robotic Automation and load the cell project without the part.

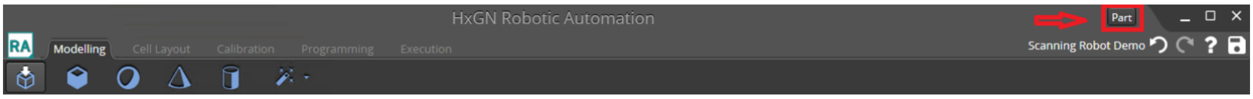
2) Import the Part CAD Model in HxGN Robotic Automation application. User may also skip this step and import the CAD and features in Inspire first; CAD and features will then be automatic transferred to the HxGN Robotic Automation.

3) Align the part in the cell and attach it to the fixture if needed. Once part is placed on the HxGN Robotic Automation Cell Project and attached to the Fixture, press on the metrology traffic light to open Inspire. Automatically Inspire opens and the Part CAD is loaded in Inspire. If the part was not imported in HxGN Robotic Automation first, it will be imported after the connection with Inspire is established.

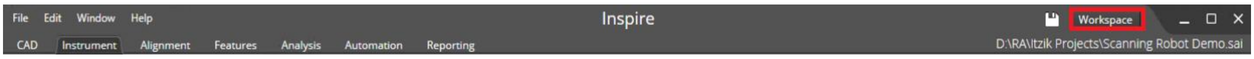




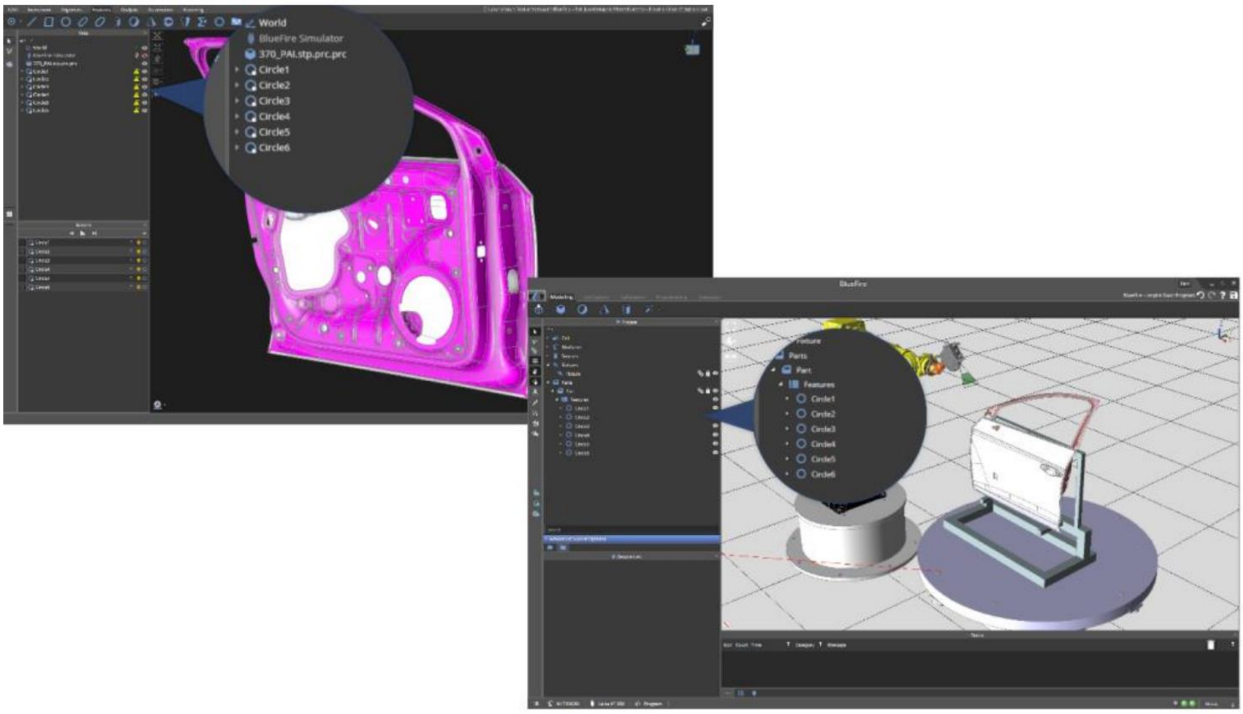
4) Save the Project in HxGN Robotic Automation, use the **Part** button to switch from HxGN Robotic Automation to Inspire.



5) Automatically Inspire saves the project created in Inspire with the same name used in HxGN Robotic Automation but with *.sai extension. Within Inspire use the **Workspace** button to switch from Inspire to HxGN Robotic Automation.

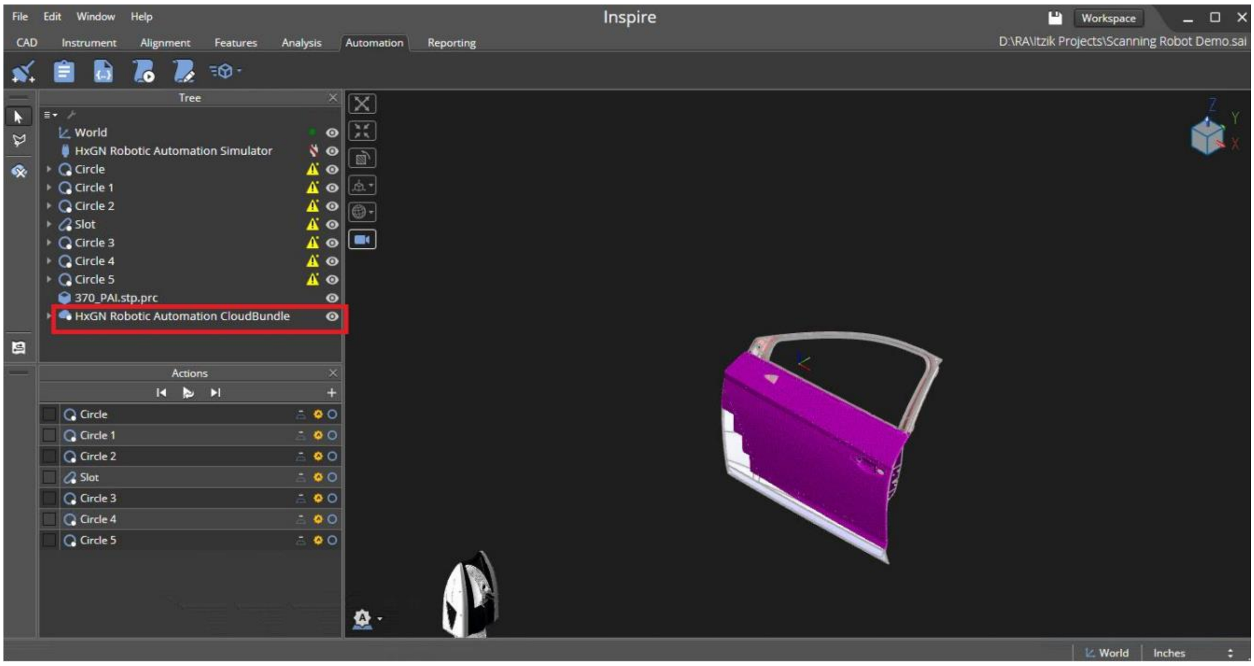


6) Add in Inspire the features that user wants to inspect. These will be automatically sent to HxGN Robotic Automation and added to the Part Features List.

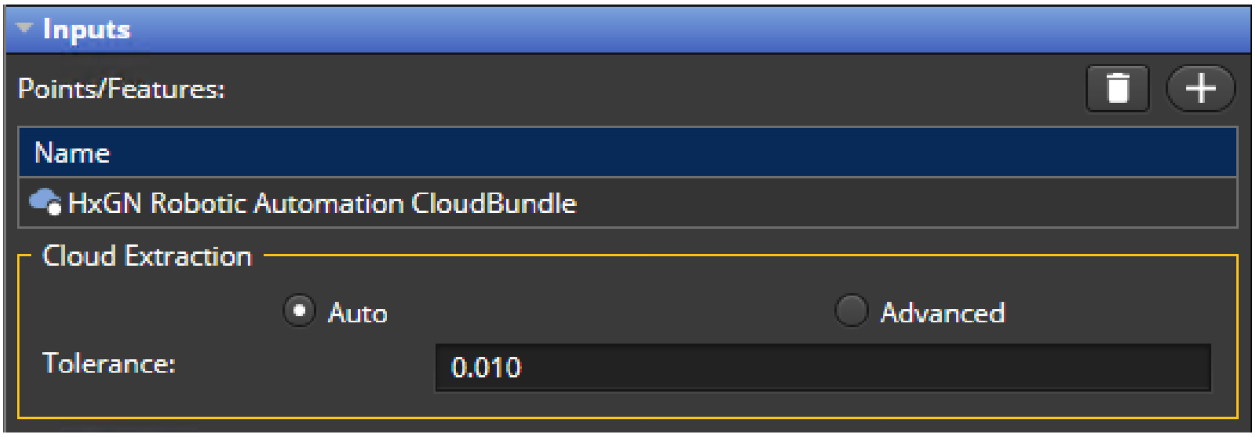


7) Once the Inspire program has all the features needed for the part inspection, the user is ready to prepare the robot program in HxGN Robotic Automation.

8) When the robot program is executed from HxGN Robotic Automation, the COP data is sent to Inspire and stored in the **HxGN Robotic Automation CloudBundle** object.



9) User must make sure that the properties of the features inserted from the CAD file in Inspire are set to use the HxGN Robotic Automation CloudBundle as the Input for the Feature Extraction.



Inputs

Points/Features:

Name
HxGN Robotic Automation CloudBundle

Cloud Extraction

● Auto ○ Advanced

Tolerance: 0.010

HxGN Robotic Automation project and the Inspire program should be stored in the same folder.

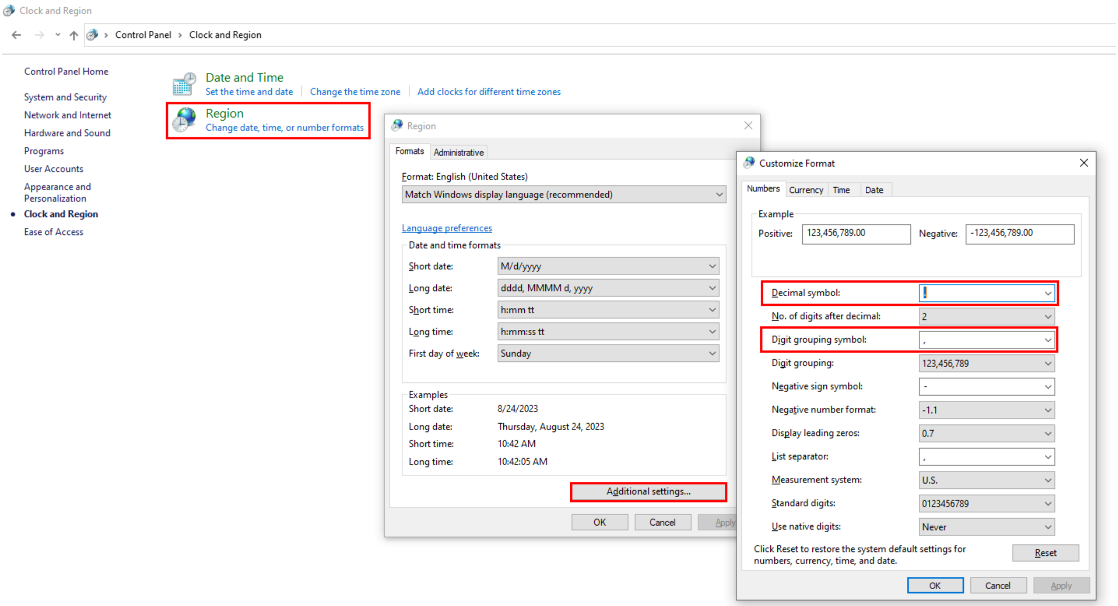
If both applications are opened and connected, edit or updating features on the Inspire program will automatically update the features on HxGN Robotic Automation project and vice versa.

Features created in HxGN Robotic Automation are not sent to Inspire. Inspire is the master on the Inspection plan side and if the user wants to include features to ease the programming task but which are not relevant for metrology computation, these can be inserted in HxGN Robotic Automation and will not be added to the Inspire Project.

10.3 PolyWorks

HxGN Robotic Automation 2023.1 is compatible with PolyWorks Metrology Suite 2023 IR1.
There is no need to change “RAConfig.xml” file as for PC-DMIS or Inspire. The user just need to configure PolyWorks as the metrology application in use within HxGN Robotic Automation project properties.

Warning: be sure that in *Control Panel* | *Clock and Region* clicking on *Region* | *Additional settings…*, the **Decimal symbol** is set to “**.**” (**dot**), and the **Digit grouping symbol** is set to “**,**” (**comma**), as shown in the picture below.



Format: English (United States)
Match Windows display language (recommended)

Date and time formats
Short date: M/d/yyyy
Long date: dddd, MMMM d, yyyy
Short time: h:mm tt
Long time: h:mm:ss tt
First day of week: Sunday

Examples
Short date: 8/24/2023
Long date: Thursday, August 24, 2023
Short time: 10:42 AM
Long time: 10:42:05 AM

Example
Positive: 123,456,789.00
Negative: -123,456,789.00

Decimal symbol: .
No. of digits after decimal: 2
Digit grouping symbol: ,
Digit grouping: 123,456,789
Negative sign symbol: -
Negative number format: -1.1
Display leading zeros: 0.7
List separator: ,
Measurement system: U.S.
Standard digits: 0123456789
Use native digits: Never

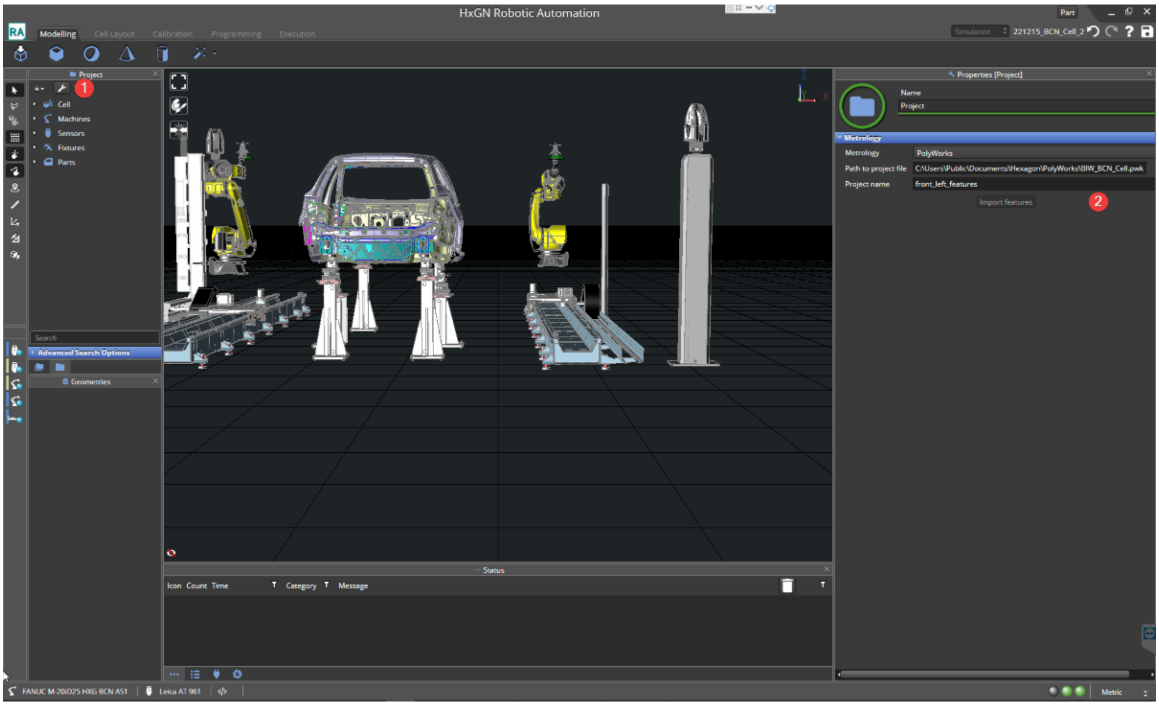
Click Reset to restore the system default settings for numbers, currency, time, and date.

10.3.1 Workflow using PolyWorks

The user must open a new project in HxGN Robotic Automation and import the CAD manually into HxGN Robotic Automation as automatic CAD import from PolyWorks is not supported.
Under the project tree the user must:

1) Click on the **Project Properties** icon to open the properties window on the right and select PolyWorks from the Metrology drop-down menu.
2) Set the **Path to project file** and PolyWorks **Project name** to be used.

Warning: this filename needs to be within the metrology folder defined in the settings (see chapter 4.11.1.1) as it will be saved as a relative path to that folder!



Properties [Project]

Name: Project

Metrology

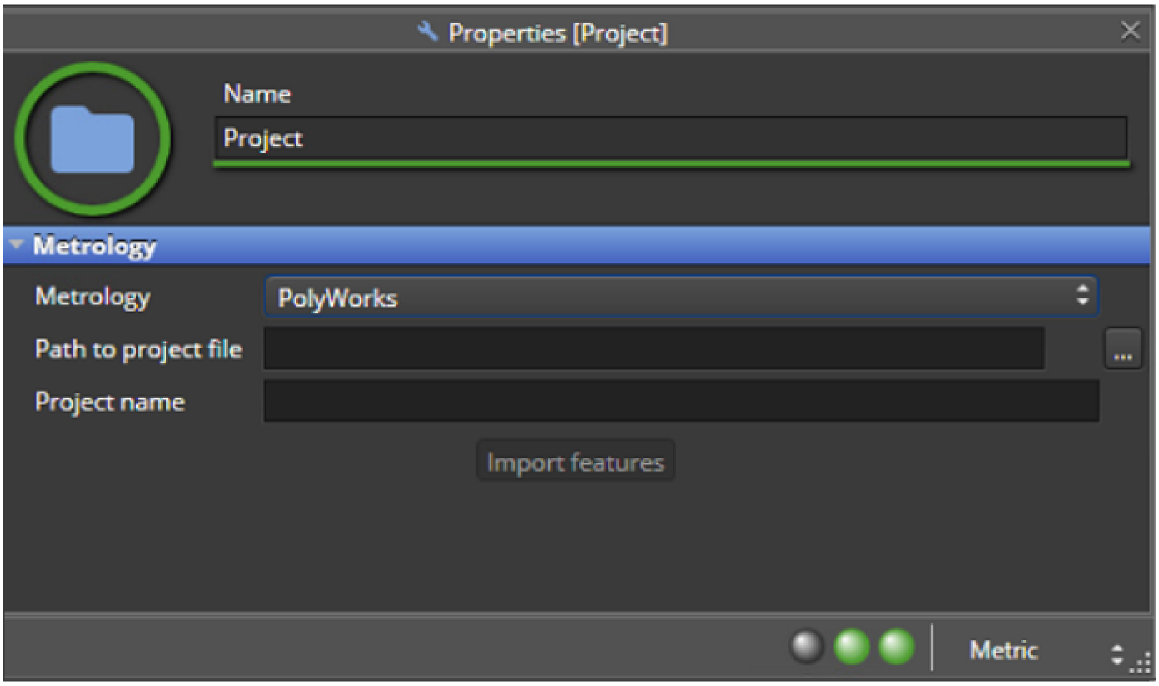
Metrology: PolyWorks

Path to project file: C:\Users\Public\Documents\Hexagon\PolyWorks\BIW_BCN_Cell.pwk

Project name: front_left_features

Import features

Pay attention that PolyWorks structure has a Project file, and under it there is a Project name. Importing features capability is disabled when connection to metrology is disabled (Metrology Activation button is grey).



Name: Project

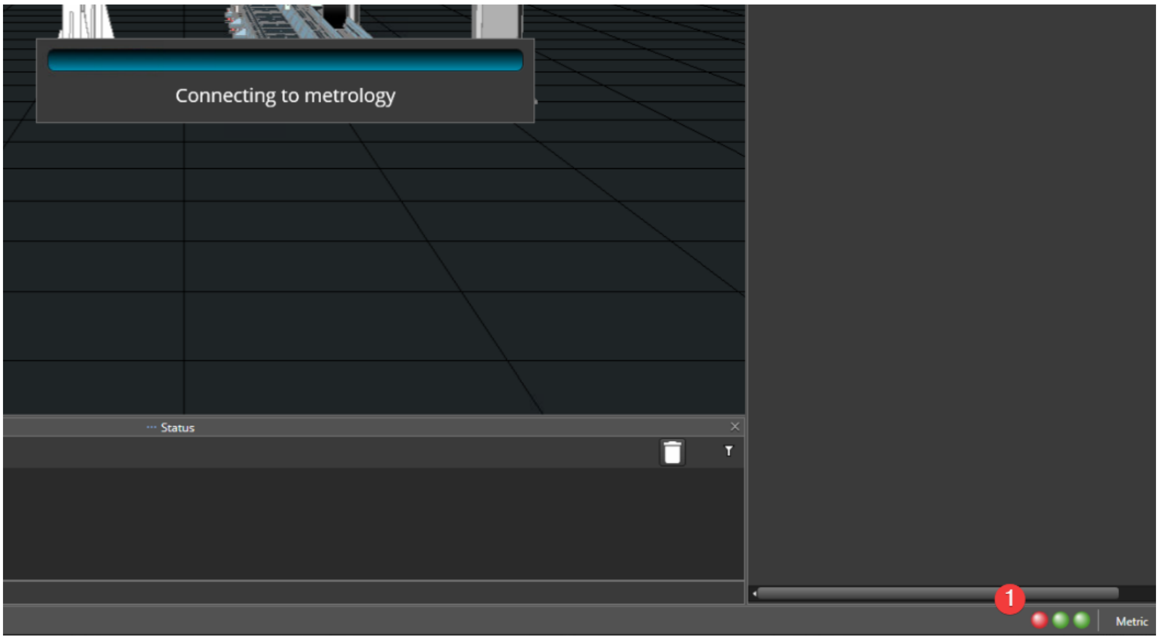
Metrology

Metrology: PolyWorks

Path to project file:

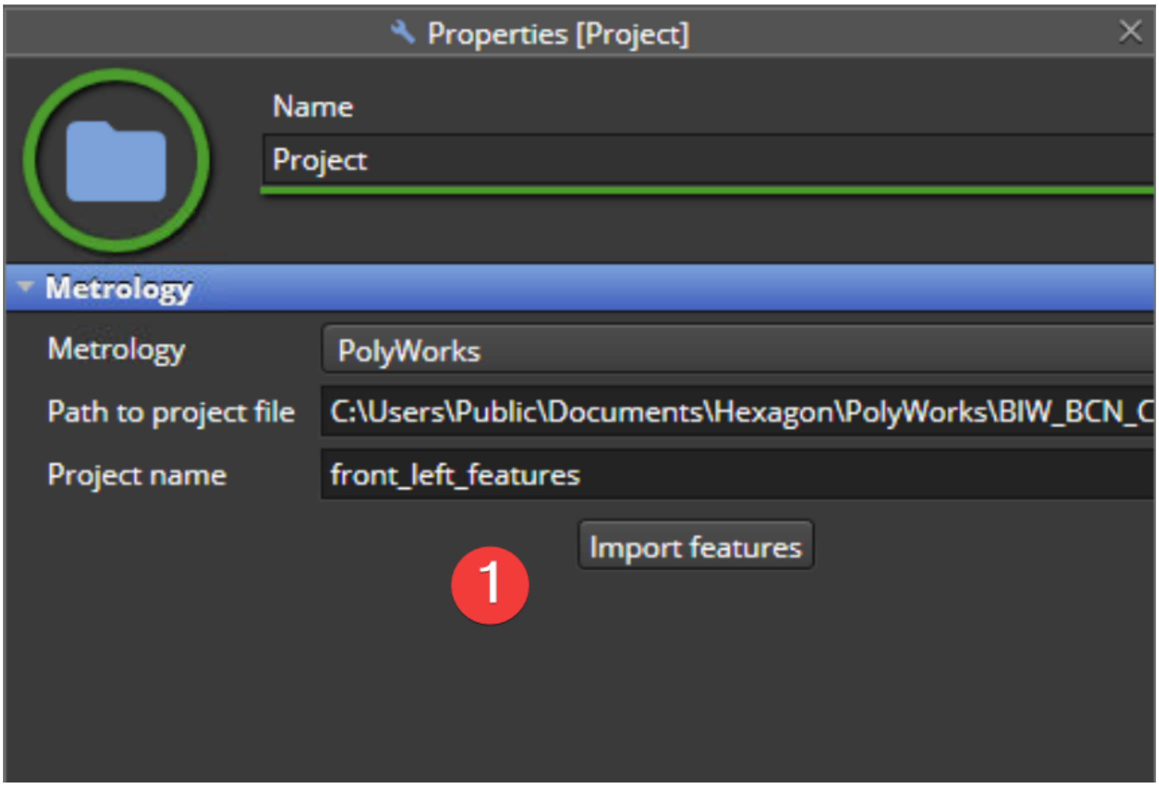
Project name:

3) Connect to PolyWorks: the Metrology button will turn to green once the connection is established.



4) Now the user can import the features in HxGN Robotic Automation, and it will be possible to start programming the robot path to scan them. A list of supported features can be found in chapter 0.

Warning: Ensure to use the English language in the PolyWorks project while creating the features, otherwise the import in HxGN Robotic Automation may fail.


Properties [Project]

Name: Project

Metrology

Metrology: PolyWorks

Path to project file: C:\Users\Public\Documents\Hexagon\PolyWorks\BIW_BCN_C

Project name: front_left_features

5) Once the program has been written in HxGN Robotic Automation, the user can execute the program either on HxGN Robotic Automation or Execution Manager (see chapter 9.2).

   a) Executing in HxGN Robotic Automation:

      i) If the PolyWorks Inspector program is already opened, HxGN Robotic Automation will send the COP to this one.

      ii) If the PolyWorks Inspector program is not opened, HxGN Robotic Automation will open PolyWorks Inspector program and send the COP to that instance.

Info: when importing a project in HxGN Robotic Automation, all the Macro files that are used will need to be seperately copied to: **<Metrology Folder>/<Project name>/<Program name>**

   b) Executing in Execution Manager:

      i) make sure that **MetrologyFolderPath** under **ExecutionManagerSettings.json** file is defined exactly as in HxGN Robotic Automation.

      ii) In case Generic Host is running as a console application, PolyWorks Inspector behaves as in HxGN Robotic Automation.

      iii) In case Generic Host is running as a service, PolyWorks Inspector is hidden. Everything works in the background as a process without the visual application running.

10.3.1.1 Programming PolyWorks Macros

The user can add an **Execute PolyWorks Macro** command (⚙) to the program to run specific execution routines on PolyWorks. Once added to the program the user can specify the properties.

HxGN Robotic Automation offers pre-made macros (like **Set Device Position** to be used with reference points as explained in chapter 10.3.1.5) or custom macros, in which the user can add commands as needed.

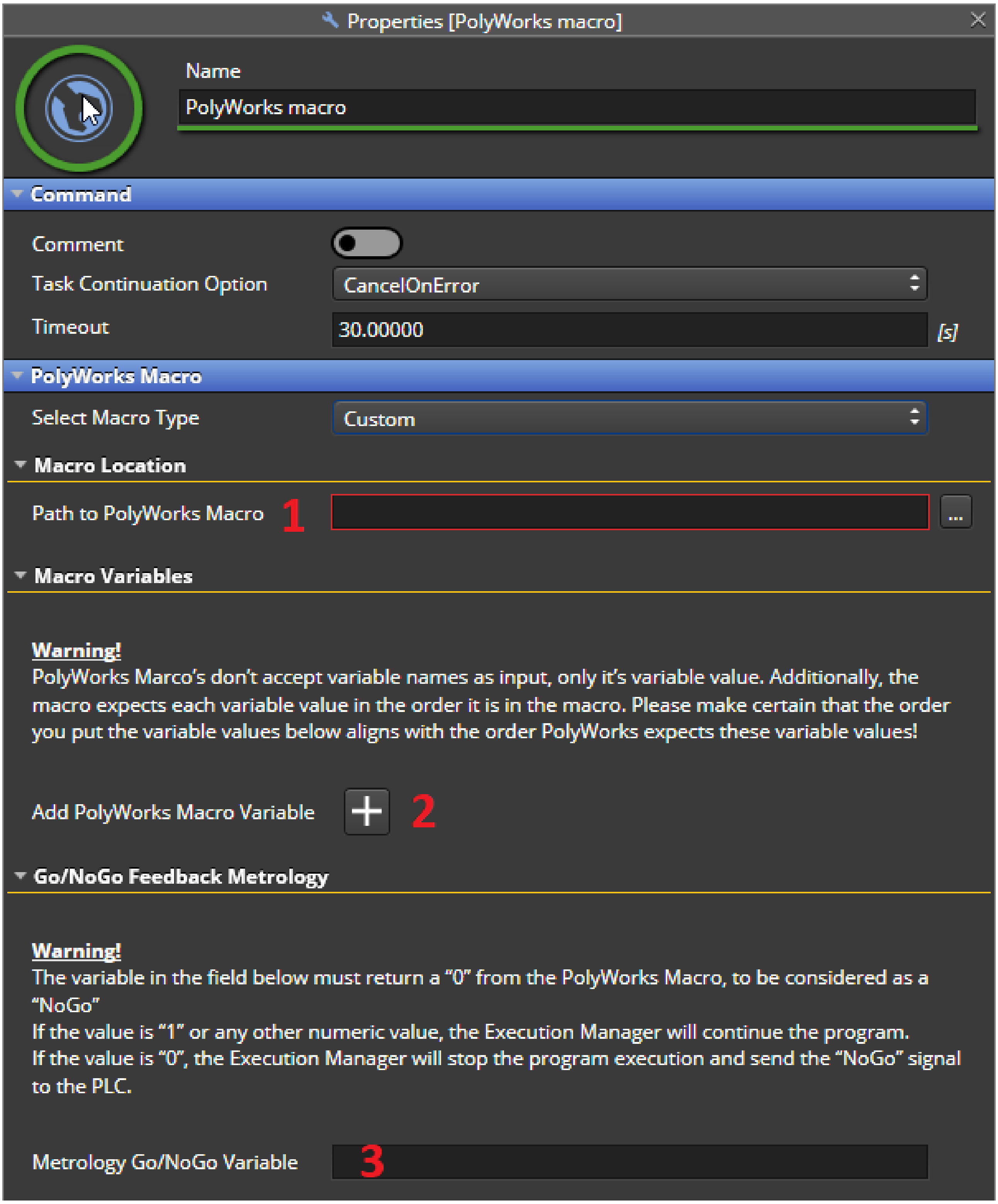
After selecting Custom within the Select Macro Type drop-down menu, the following properties will be shown:

1) Under *PolyWorksMacro* | *Macro Location* in **Path to PolyWorks Macro** field the user must store the PW Macro.

Warning: This Macro needs to be within the metrology folder defined above as it will be saved as a relative path to that folder!

2) Under *PolyWorksMacro* | *Macro Variables* the user can set a specific variable value for the Macro. PolyWorks expects to receive these values according to the order they are listed: the first value will be evaluated for the first variable that is expecting a value, the second value will be evaluated for the second variable that is expecting a value, and so on.

3) Under *PolyWorksMacro* | *Go/NoGo Feedback Metrology* the user can set a Go/NoGo variable. Execution Manager will check the value of this variable as a feedback from the PolyWorks Macro Command. Refer to chapter for further information on the usage.



Properties [PolyWorks macro]

Name: PolyWorks macro

Command

Comment: ●

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]

PolyWorks Macro

Select Macro Type: Custom

Macro Location

Path to PolyWorks Macro 1

Macro Variables

Warning!
PolyWorks Marco's don't accept variable names as input, only it's variable value. Additionally, the macro expects each variable value in the order it is in the macro. Please make certain that the order you put the variable values below aligns with the order PolyWorks expects these variable values!

Add PolyWorks Macro Variable + 2

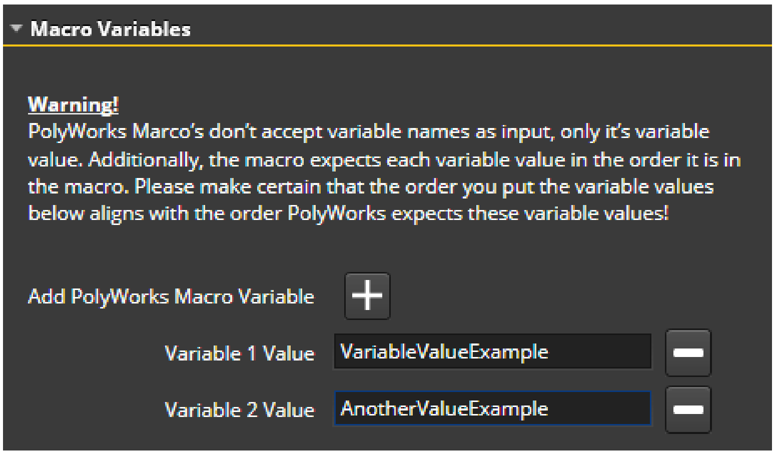
Go/NoGo Feedback Metrology

Warning!
The variable in the field below must return a "0" from the PolyWorks Macro, to be considered as a "NoGo"
If the value is "1" or any other numeric value, the Execution Manager will continue the program.
If the value is "0", the Execution Manager will stop the program execution and send the "NoGo" signal to the PLC.

Metrology Go/NoGo Variable 3

Example – Using variable values.

Two variable values have been added: *VariableValueExample* and *AnotheValueExample*.



Macro Variables

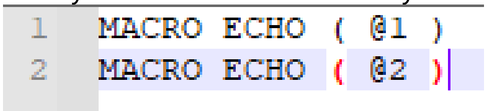
Warning!
PolyWorks Marco's don't accept variable names as input, only it's variable value. Additionally, the macro expects each variable value in the order it is in the macro. Please make certain that the order you put the variable values below aligns with the order PolyWorks expects these variable values!

Add PolyWorks Macro Variable

Variable 1 Value: VariableValueExample

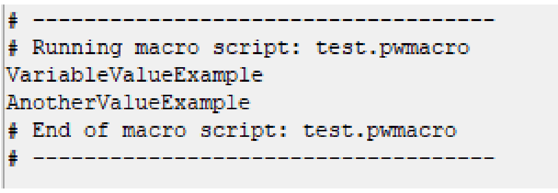
Variable 2 Value: AnotherValueExample

In PolyWorks to call those values you would need to use @1 for the first and @2 for the second value.



```
MACRO ECHO ( @1 )
MACRO ECHO ( @2 )
```

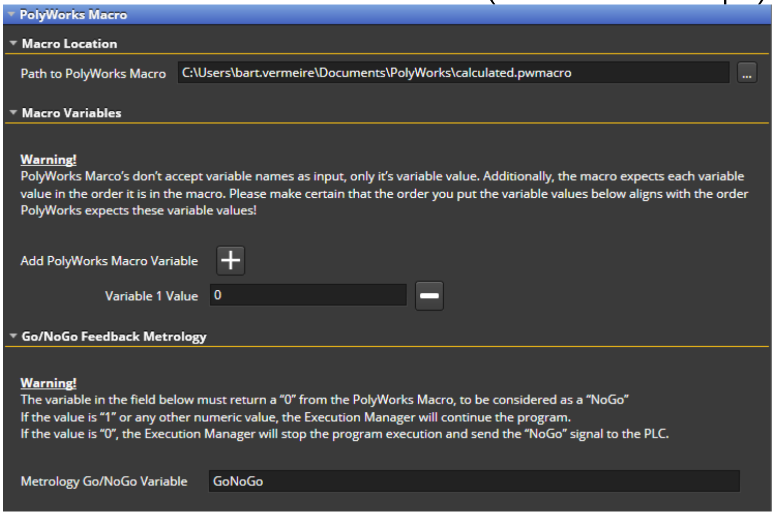
This will result in PolyWorks as:



```
# ------------------------------------
# Running macro script: test.pwmacro
VariableValueExample
AnotherValueExample
# End of macro script: test.pwmacro
# ------------------------------------
```

Example – Using Go/NoGo Feedback

Create a macro that will feedback failure (hardcoded in example).



PolyWorks Macro

Macro Location

Path to PolyWorks Macro: C:\Users\bart.vermeire\Documents\PolyWorks\calculated.pwmacro

Macro Variables

Warning!
PolyWorks Marco's don't accept variable names as input, only it's variable value. Additionally, the macro expects each variable value in the order it is in the macro. Please make certain that the order you put the variable values below aligns with the order PolyWorks expects these variable values!

Add PolyWorks Macro Variable

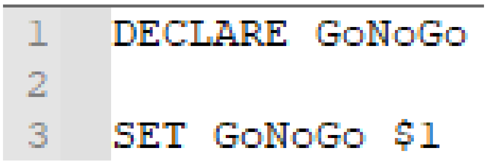
Variable 1 Value: 0

Go/NoGo Feedback Metrology

Warning!
The variable in the field below must return a "0" from the PolyWorks Macro, to be considered as a "NoGo"
If the value is "1" or any other numeric value, the Execution Manager will continue the program.
If the value is "0", the Execution Manager will stop the program execution and send the "NoGo" signal to the PLC.

Metrology Go/NoGo Variable: GoNoGo

In PolyWorks have the following Macro:



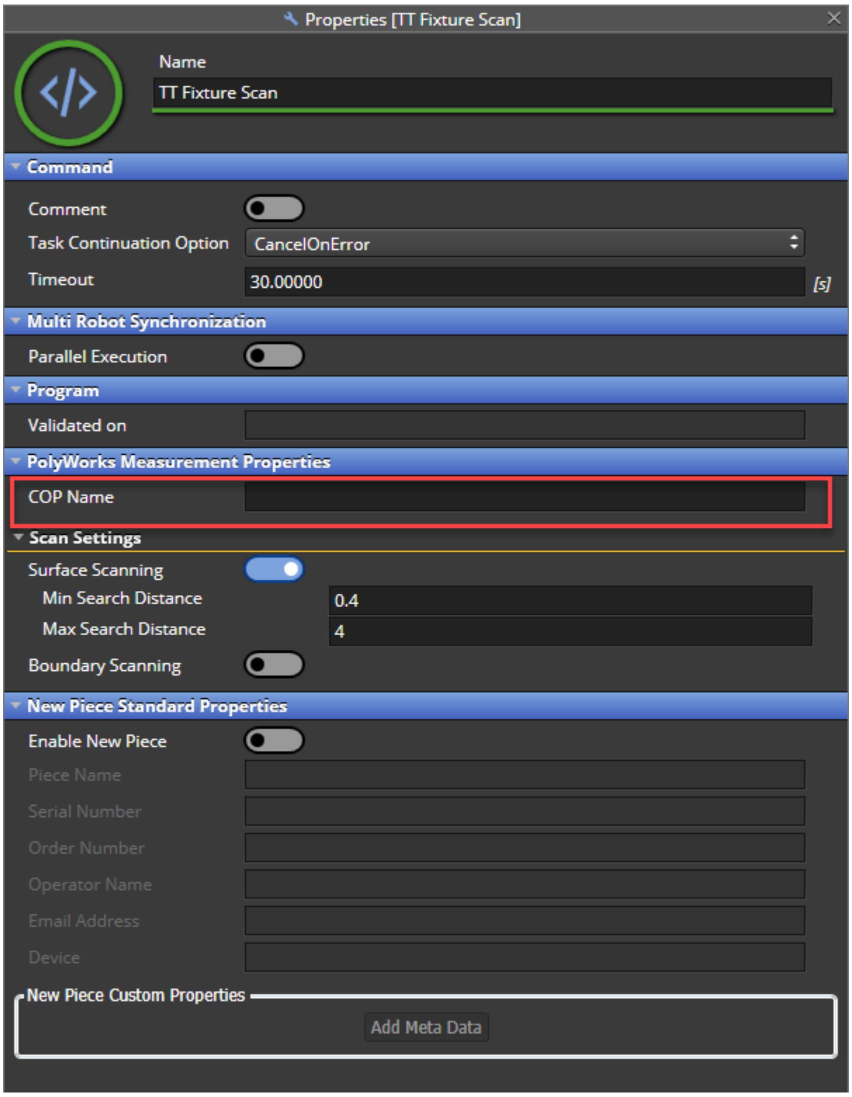
```
DECLARE GoNoGo

SET GoNoGo $1
```

Note: in a real Go/NoGo scenario the value will be decided by PolyWorks based on the feature calculation and not as in this example based on the Variable 1 Value.

10.3.1.2 Named COP

It is possible to define a global COP name to be used throughout the entire program execution or to define local COPs to be sent as separate data sets at every scan.

In the first case, the global COP name can be specified in the Program properties under *COP* | *COP Name*. If no name is specified, the COP will be named “Ra Cop.



Properties [TT Fixture Scan]

Name: TT Fixture Scan

Command
- Comment: ○
- Task Continuation Option: CancelOnError
- Timeout: 30.00000 [s]

Multi Robot Synchronization
- Parallel Execution: ○

Program
- Validated on:

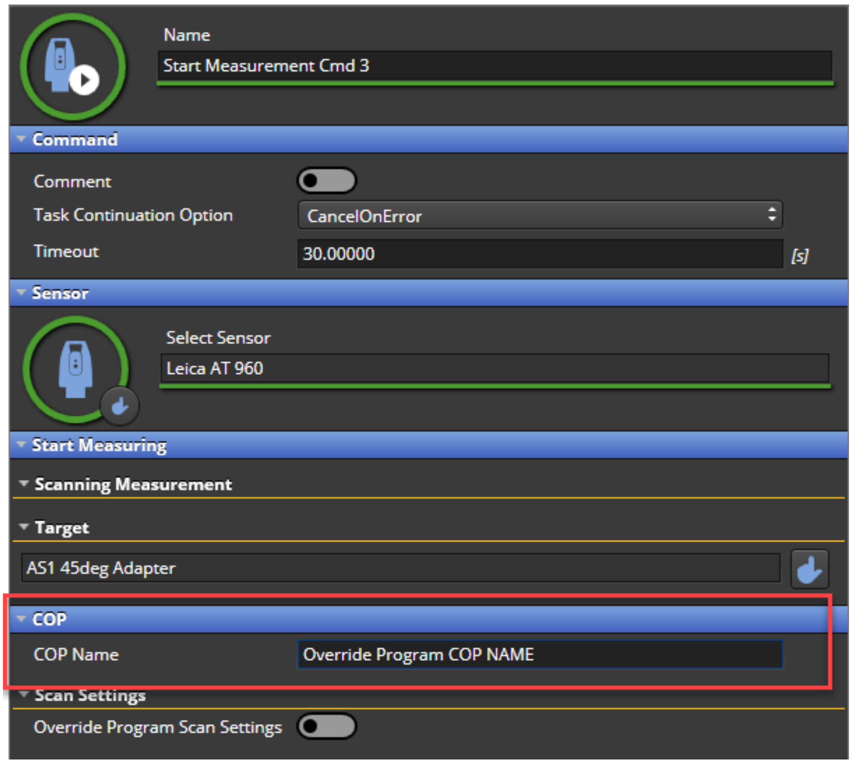
PolyWorks Measurement Properties
- COP Name:

Scan Settings
- Surface Scanning: ●
  - Min Search Distance: 0.4
  - Max Search Distance: 4
- Boundary Scanning: ○

New Piece Standard Properties
- Enable New Piece: ○
- Piece Name
- Serial Number
- Order Number
- Operator Name
- Email Address
- Device

New Piece Custom Properties

In the second case, the local COP name can be specified in the Start Measurement command properties also under *COP | COP Name.*



Name: Start Measurement Cmd 3

Command

Comment: ○

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]

Sensor

Select Sensor: Leica AT 960

Start Measuring

Scanning Measurement

Target: AS1 45deg Adapter

COP

COP Name: Override Program COP NAME

Scan Settings

Override Program Scan Settings: ○

If a Program Properties COP has been defined, this will be the name of the COP for the whole program, unless a different name is specified within another Start Measurement command property.

Info: in case of multiple iterations of the same program for the same piece, please avoid having more than one space character in the COP name. This will cause the COP on subsequent iterations of the program to create a new COP container in PolyWorks, rather than append to the existing one. E.g., if COP Name is: “Test 1 Container” (containing two spaces) on the second run it will create a new container “Test 1 Container(1)” rather than append to the original one.

The COP Name specified in the Start Measurement command properties will overwrite the one defined under Program Properties. If nothing is specified as COP Name neither in the first nor in the second scenario, the resulting COP will be called “Ra Cop”.

Example

Program COP: Name_Example1

- Start measurement1 left blank
- Start measurement2: Named: Name_Example2
- Start measurement3 left blank
- Start measurement4: Named: Name_Example3

- Start measurement5: Named: Name Example2

Will results in:

- COP: Name_Example1 consists out of
  - Start measurement1
  - Start measurement3
- COP: Name_Example2 consists out of:
  - Start measurement2
  - Start measurement5
- COP: Name_Example3 consists out of:
  - Start measurement4

10.3.1.3 Scan Settings

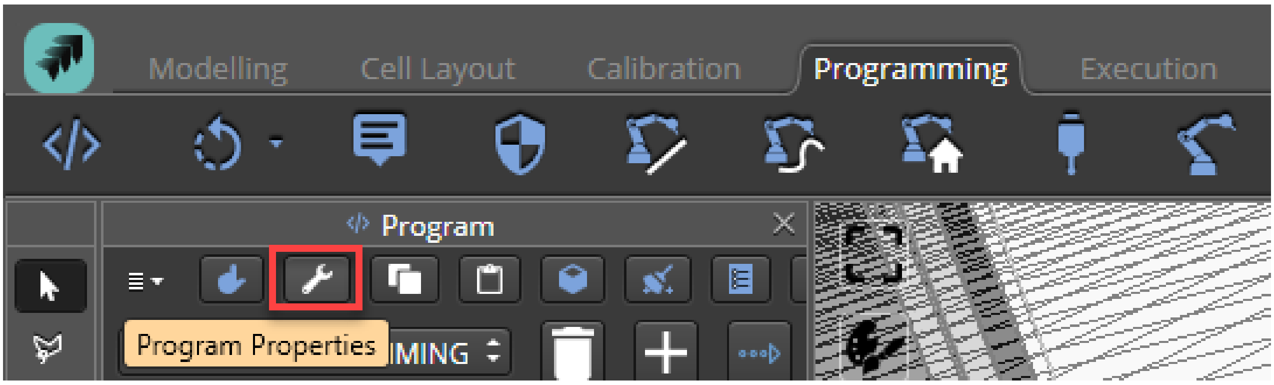
By default a program created under a PolyWorks enabled project has the Surface Scan enabled with the following default parameters:

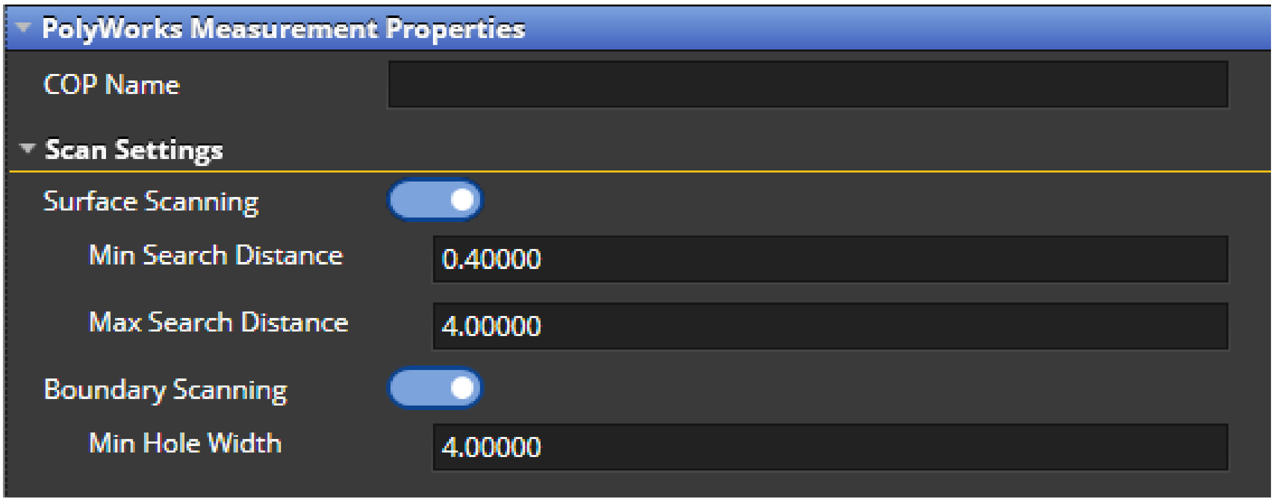
- Min. Search Distance: 0.40 mm
- Max. Search Distance: 4.0 mm

The boundary scan when enabled will have the following default parameter:

- Min. Hole Width: 4.0 mm

The settings can be changed for the whole program by going to the **Program Properties** of a PolyWorks enabled project.





PolyWorks Measurement Properties

COP Name

Scan Settings

Surface Scanning ●

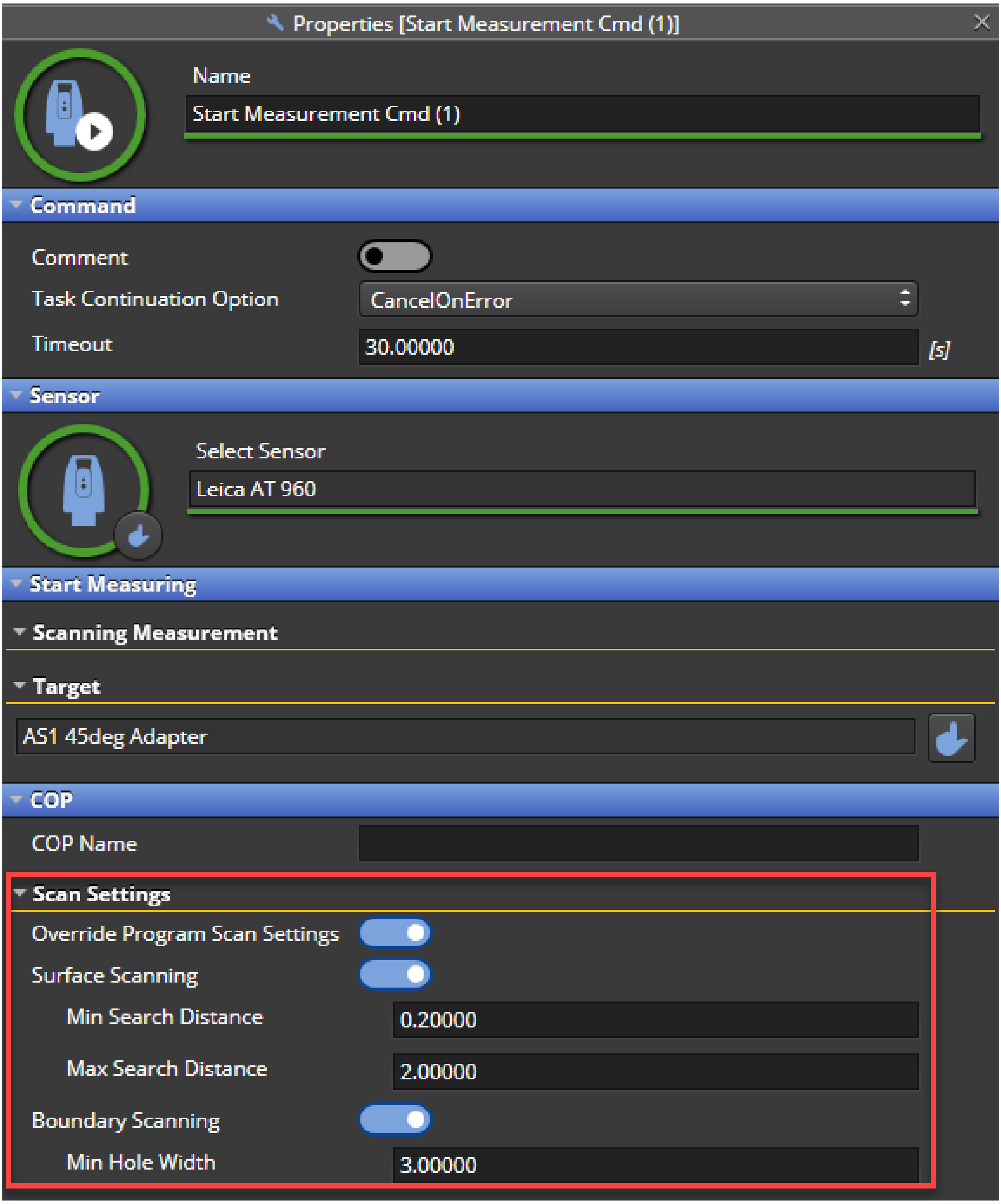
Min Search Distance 0.40000

Max Search Distance 4.00000

Boundary Scanning ●

Min Hole Width 4.00000

Each individual start measurement command can overwrite the **Scan Settings** between that start measurement command and the next stop measurement command.



Properties [Start Measurement Cmd (1)]

Name: Start Measurement Cmd (1)

Command

Comment: ○

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]

Sensor

Select Sensor: Leica AT 960

Start Measuring

Scanning Measurement

Target

AS1 45deg Adapter

COP

COP Name:

Scan Settings

Override Program Scan Settings: ●

Surface Scanning: ●

Min Search Distance: 0.20000

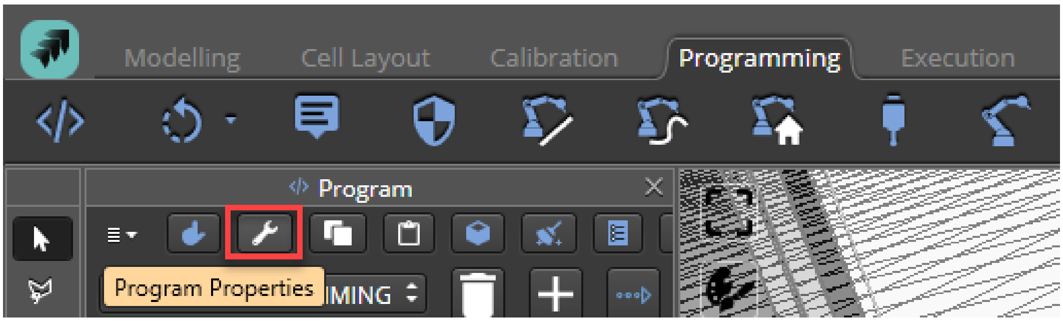
Max Search Distance: 2.00000

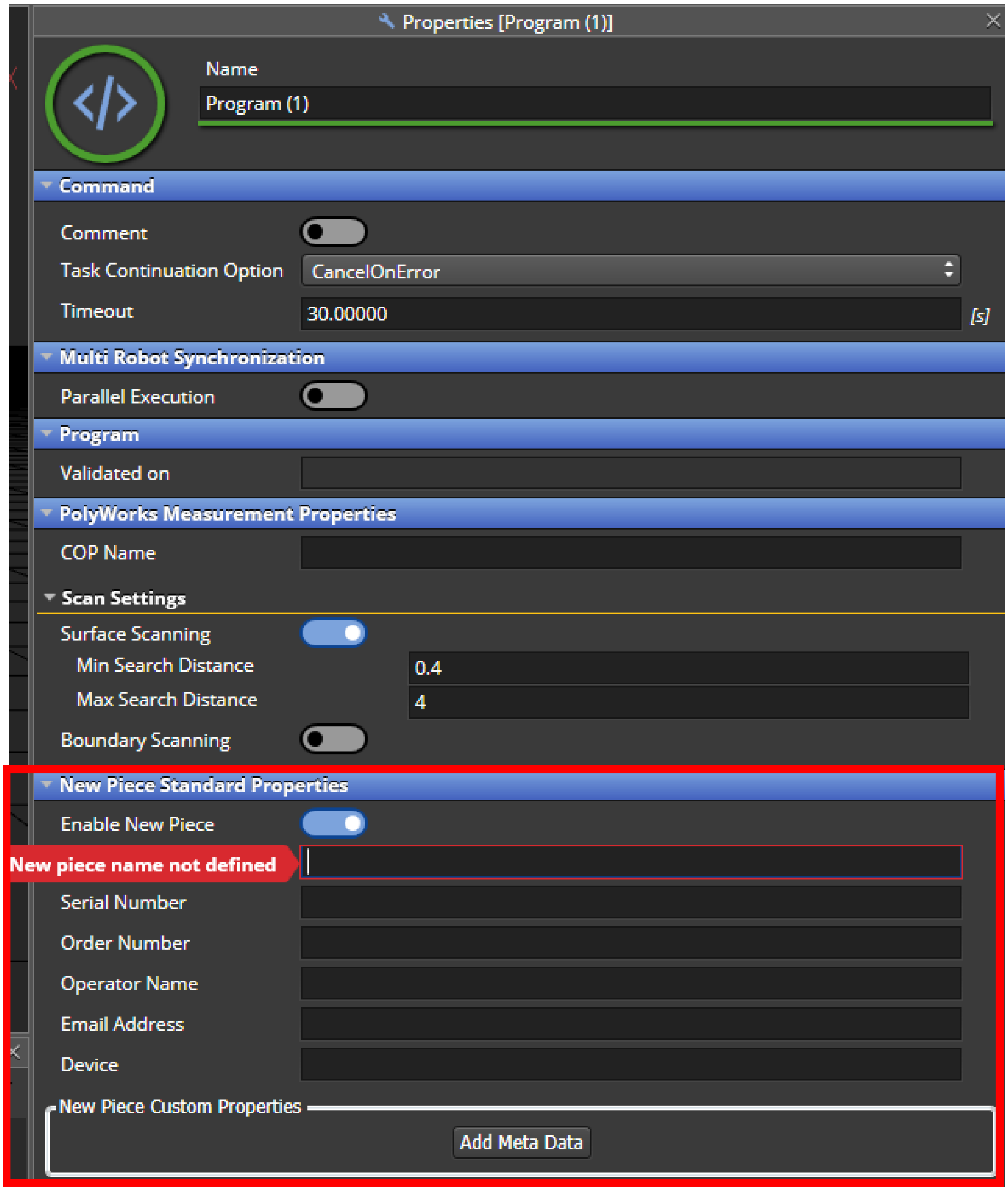
Boundary Scanning: ●

Min Hole Width: 3.00000

10.3.1.4 Setting metadata

To enable the new piece creation in PolyWorks, go to the program properties and select “Enable New Piece”.




Properties [Program (1)]

Name: Program (1)

Command

Comment: ○

Task Continuation Option: CancelOnError

Timeout: 30.00000 [s]

Multi Robot Synchronization

Parallel Execution: ○

Program

Validated on:

PolyWorks Measurement Properties

COP Name:

Scan Settings

Surface Scanning: ●

Min Search Distance: 0.4

Max Search Distance: 4

Boundary Scanning: ○

New Piece Standard Properties

Enable New Piece: ●

New piece name not defined

Serial Number:

Order Number:

Operator Name:

Email Address:

Device:

New Piece Custom Properties

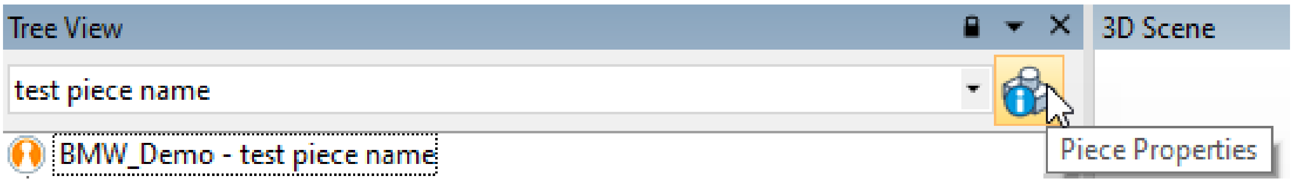
Add Meta Data

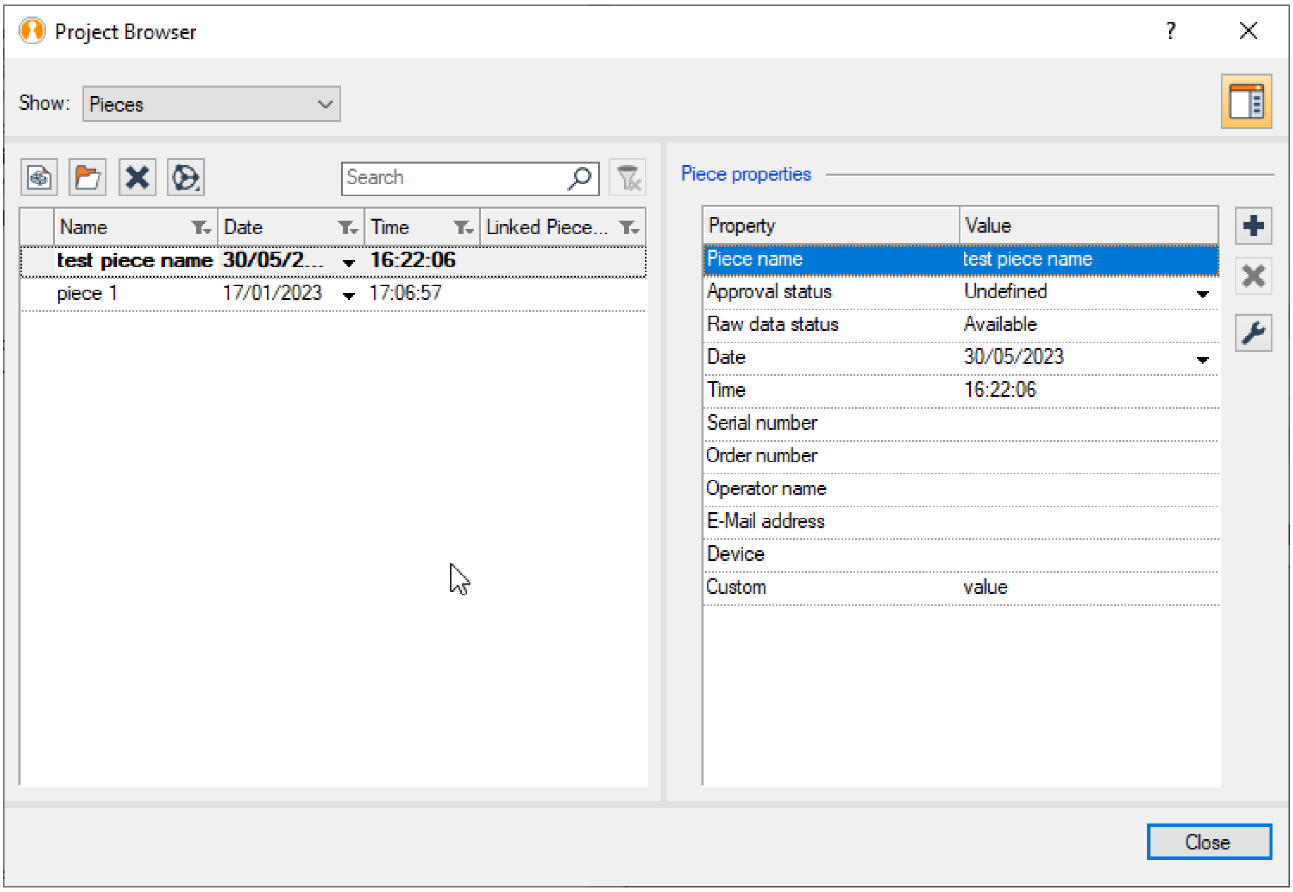
The new piece is not enabled by default in order to allow the user to run various programs for only one piece in PolyWorks. Once enabled, the Piece Name must be specified.

HxGN Robotic Automation will already provide some fields that are default piece properties in PolyWorks, while also providing an option to add custom Meta Data.

Info: When creating a new piece, scan data is saved automatically within PolyWorks. In case the program is not configured to create a new piece, this data will have to be saved manually or through a custom macro.

In PolyWorks, metadata can be visualized by clicking on the Piece Properties.





Name	Date	Time	Linked Piece...
test piece name	**30/05/2...**	**16:22:06**	
piece 1	17/01/2023	17:06:57	

Property	Value
Piece name	test piece name
Approval status	Undefined
Raw data status	Available
Date	30/05/2023
Time	16:22:06
Serial number	
Order number	
Operator name	
E-Mail address	
Device	
Custom	value

10.3.1.5 Reference points

Reference points are used in PolyWorks to compensate the cloud of point as it is done using internal HxGN Robotic Automation compensation.
The functionality is similar to PC-DMIS Static points, used for part alignment workflow with PC-DMIS (see 10.1.2). Reference points workflow with PolyWorks consists of HxGN Robotic Automation sending reflector positions to PolyWorks so it can either use them to align the various COPs received during the scans or to check part alignment.

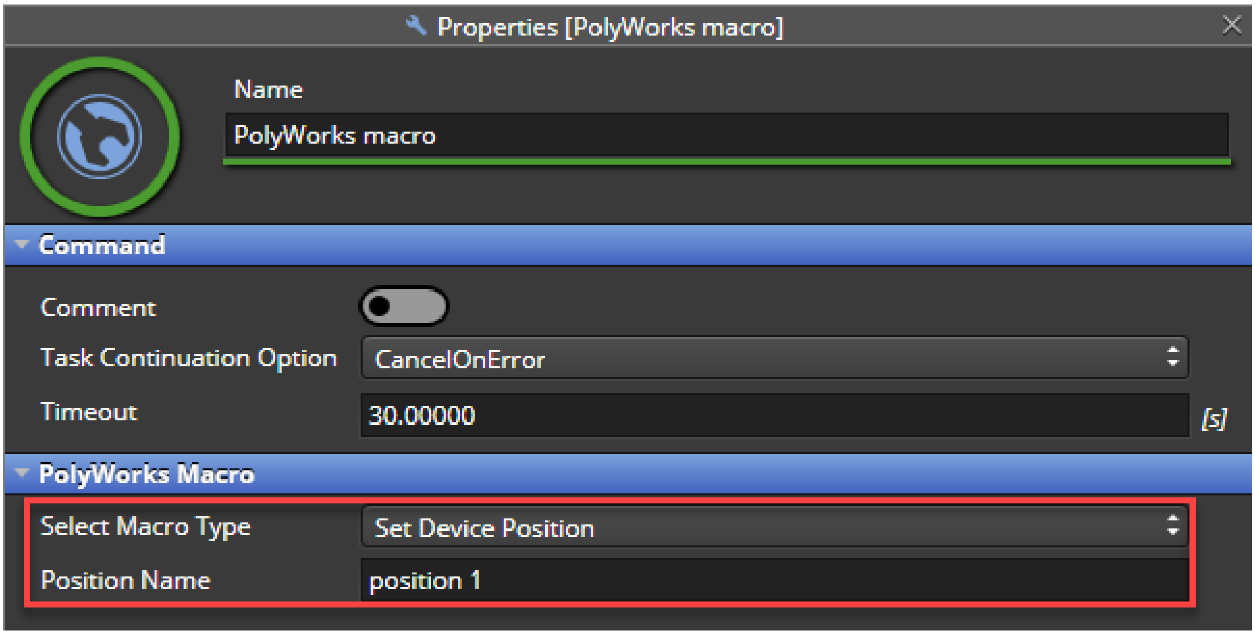
Warning: when using reference points, COP will appear in a wrong place in RA because data compensation is managed in PolyWorks side.

To set the reference points, it is first necessary to:

1) Create device positions (if they do not already exist) in the PolyWorks project to be used either using Macro programming directly in PolyWorks or using an Execute Macro command within the HxGN Robotic Automation program.

2) Set the active device position by using the predefined **Set Device Position** macro available in HxGN Robotic Automation. This predefined macro allows the software to automatically ensure the COP name is not re-used in different positions:

   a) add an **Execute PolyWorks Macro** command
   b) select **Set Device Position** as Macro Type from the relative drop-down menu
   c) set the device position name (case sensitive) as defined in the relative PolyWorks project (e.g. “position 1”)



Properties [PolyWorks macro]

Name: PolyWorks macro

Command

Comment: ○
Task Continuation Option: CancelOnError
Timeout: 30.00000 [s]

PolyWorks Macro

Select Macro Type: Set Device Position
Position Name: position 1

The workflow using Reference points within an HxGN Robotic Automation program is shown in the following two examples below.

Example – Rotary table moving to different scan positions

1) Turn table and Robots move to home positions
2) Create the positions in PolyWorks through a macro in case they don't exist yet in the project.
3) Set Device Position Macro
4) Measure the reflectors
5) Measure a COP
6) Rotate the turn table
7) Set Device Position Macro
8) Measure the reflectors
9) Measure a COP
10) Repeat steps 6) to 9) to add more rotary table scan positions

Example – Tracker slider moving to different scan positions

1) Tracker slider and Robots move to home positions
2) Create the positions in PolyWorks through a macro in case they don't exist yet in the project.
3) Set Device Position Macro
4) Measure the reflectors
5) Measure a COP
6) Move the tracker slider
7) Set Device Position Macro
8) Measure the reflectors
9) Measure a COP
10) Repeat steps 6) to 9) to add more tracker slider scan positions

Info: The order of either doing first the COP measurement (step 5) or the reflector measurement (step 4) is not important.

Warning: Due to a PolyWorks limitation it is required that the COP name's cannot be re-used in different positions. E.g.:
Position1
Start measurement (COP NAME: BIW_LEFT)
Start measurement (COP NAME: BIW_RIGHT)
Position2
Start measurement (COP NAME: BIW_LEFT) → **Error!**

Warning: PolyWorks does not support multiple positions at the same time. This means reference points cannot be used for a dual robot, dual tracker cell where it is needed to have two parallel scans at the same time. Using HxGN Robotic Automation compensation (Nominal vs Actual reflector positions, see chapter 8.7) is recommended for this use case.

10.4 List of supported features

The following table shows how the different metrology software to which HxGN Robotic Automation interfaces with support the various feature types available.

Feature type	PC-DMIS	Inspire	PolyWorks
Circle	✓	✓	✓
Cone	✓	✓	✓
Cylinder	✓	✓	✓
Edge Line	✗	✓	✗
Edge Point	✓	✗	✓
Ellipse	✗	✗	✓
Flanged Circle	✗	✗	✗
Flanged Slot	✗	✗	✗
Hem Point	✗	✗	✓
Hexagon	✓	✗	✗
Plane	✓	✓	✗
Poly-surface	✗	✓	✗
Polyline	✗	✗	✓
Slot	✓	✓	✓(1)
Sphere	✓	✓	✓
Surface point	✓	✓	✓
Keyhole	✗	✗	✗

(1) A Rectangle feature imported from PolyWorks into HxGN Robotic Automation will be converted to a Slot.

11 FAQ

This is a collection of Frequently Asked Questions (FAQs) about HxGN Robotic Automation.

Question	Answer
Which Robots have a direct Interface with HxGN Robotics Automation?	FANUC robots using R-30ia controllers and KUKA robots using KR C4 controllers.
Which Scanners have a direct Interface with HxGN Robotics Automation?	HxGN Robotics Automation supports the Leica T-SCAN and HxGN AS1 Scanners. More HW components will be integrated in future versions.
Which PLC's have a direct Interface with HxGN Robotics Automation?	HxGN Robotics Automation supports SIEMENS PLC Interface through OPC UA Protocol.
Which Metrology SW has direct Interface with HxGN Robotics Automation?	HxGN Robotics Automation interfaces with HxGN PC-DMIS and Inspire Metrology SW. It also works with InnovMetric PolyWorks. More SW Packages from HxGN and 3rd party will be supported in future versions.
Which Robot postprocessors exist in HxGN Robotics Automation?	None yet.
What is a postprocessor?	A postprocessor is a unique "translator" specific to a robot controller which can convert simulated robot poses into a specific brand language that can be then loaded to the respective robot controller.
What can be done on different Robot types with no direct Interface?	If a postprocessor exists, the respective robot program can be generated and loaded to the robot controller.
What is Robot Simulation in HxGN Robotics Automation good for?	HxGN Robotics Automation Simulates the robot path generated for path inspection, taking into consideration the robot volume, collisions, tool/scanner occlusion, tracker LOS (Line of Sight), sensor data simulation. All this delivers a realistic simulation of the part inspection process that can be validated offline, even before building the actual cell.
How to get a demo SW?	After a training session a demo license is delivered to HxGN employees. In case a demo license is needed for a customer, please get in contact with: corrado.pavan@hexagon.com
We have a customer with AT901 T-Mac trackers that has asked to use HxGN Robotics Automation. Any chance that we will see an update that supports the AT901?	HxGN Robotics Automation is designed to work best with a T-Mac and AT960 using the LMF interface to connect to the Leica Absolute Trackers.

How is the security granted on the software?	A cloud license is used to protect the software. That means you can run HxGN Robotics Automation on as many PCs as you want. All that is needed is to deactivate a license on one PC and reactivate it on another one. The preferred way to use it is over the internet using online access, but we also supply a way to activate the license offline.
What is the recommended training time?	HxGN Robotics Automation can typically be trained in 3 days. Prerequisites for this training are AT9x0 and basic robot knowledge. Basic Metrology SW Knowledge is recommended (PC-DMIS, Inspire or PolyWorks).
Is it possible to imagine a TCP link between HxGN Robotics Automation and other robots different than FANUC or KUKA?	A direct link to other Robots can be develop on request and will be available in upcoming versions.
Is it possible to edit robot programs in HxGN Robotics Automation that were built on the robot teach pendant?	No, currently it is not possible to import Robot programs to the HxGN Robotics Automation SW. Still, programs can be executed Step by Step, and Robot Motions can be recorded in HxGN Robotics Automation SW.
Is Automatic path planning based on CAD features supported?	Yes, HxGN Robotics Automation supports path planning based on features and on CAD entities as surfaces and edge lines.
Does HxGN Robotics Automation have a 'results viewer', or should this be done in the metrology software you choose?	No, Measurements Results should be reviewed in the Metrology Software.
Is it possible for HxGN Robotics Automation also to simulate the handling of components, such as fixtures and workpieces?	No, we are considering that functionalities for future releases.
Cable Modeling and Simulation, dress pack movement? Will it be simulated?	No, HxGN Robotics Automation does not include cable modeling, so dress pack movement is not supported.
Are there any strategies that can be used to scan distinct types of features, such as flanged holes and slots and so on?	Yes, user can define in HxGN Robotics Automation different presets or Scan Strategies for the distinctive features to be measured.
Is it planned to integrate custom communication [example VASS / VKRC]	Yes, this can be managed with the customer as a custom development project.
Does HxGN Robotics Automation directly control the scanner settings which can change with the features to be measured?	Yes, HxGN Robotics Automation controls the scanner settings, and these can be adjusted in HxGN Robotics Automation based on the feature type and feature scanning strategy.