

# RoboDK Interface

<https://robodk.com/>  
[info@robodk.com](mailto:info@robodk.com)  
+1-855-692-7772



Contents

Interface ..... 2

    Robot Panel ..... 3

    Robot Tool (TCP) ..... 4

    Reference Frame ..... 4

    Robot Target ..... 5

    Robot Configurations ..... 6

    Object Settings ..... 7

    Main Menu ..... 8

        File Menu ..... 9

        Edit Menu ..... 9

        Program Menu ..... 10

        View Menu ..... 11

        Tools Menu ..... 12

        Utilities Menu ..... 13

        Connect Menu ..... 15

        Help Menu ..... 15

# Interface

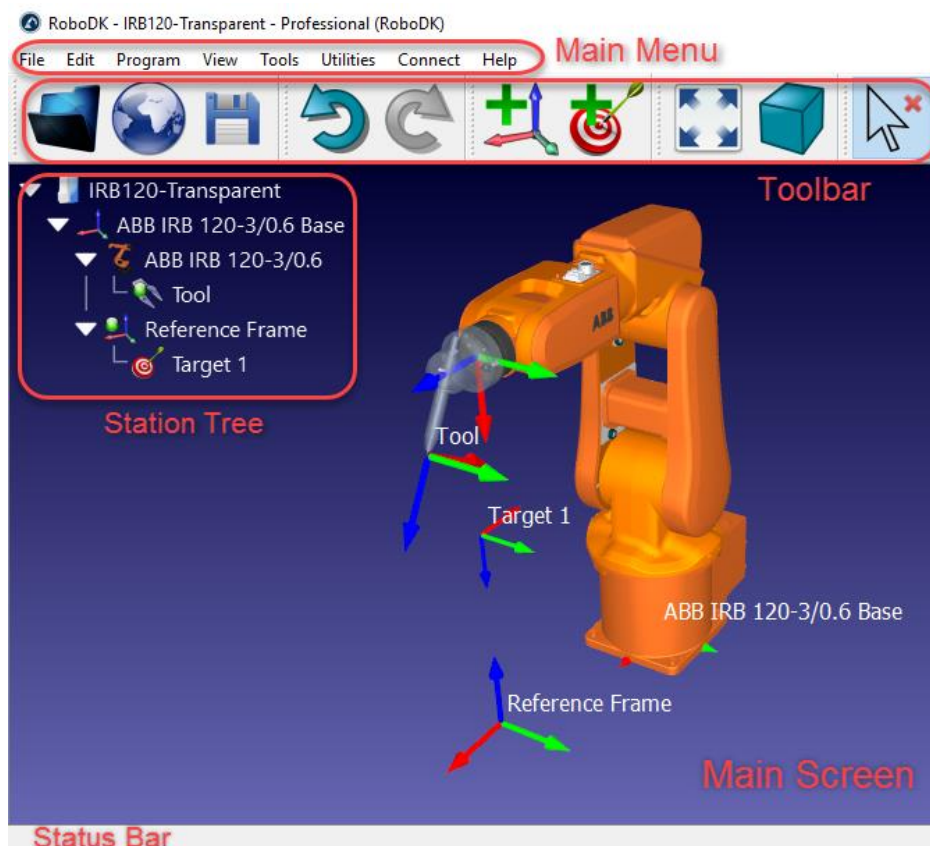
This document provides an overview of the main interface of RoboDK. The main interface is composed of the main menu, the toolbar, the station tree, the status bar and the 3D view.

- The Main Menu is located at the top. All available actions and options are available from this menu.
- The Toolbar contains graphical icons that allow quick access to frequently used actions in the menu. More information available in the Toolbar section.
- The Station Tree lists all the components (items) that are present in the station. These items can be robots, tools, objects, targets or specific settings for manufacturing or calibration purposes. The tree allows understanding and modifying the dependency that exists in the real environment. For example, a target can be attached to a specific reference frame, this reference frame can be attached to the robot base frame, robot tools are usually attached to the robot, etc.
- The Status Bar is located at the bottom and may display useful tips for certain operations.
- The 3D view (Main Screen) displays the view in a 3D virtual environment and reproduces the Station Tree with its hierarchy

**Tip:** Selecting (or double clicking) an item in the 3D view will have the same effect as selecting (or double clicking) in the item in the Station Tree and vice-versa.

Double clicking an item (in the tree or the 3D view) will display a new window with the item's properties. For example, double clicking a robot will display the [Robot Panel](#). These windows can be closed by selecting the cross at the top right of the sub window.



As an exception, double clicking a target will move the robot to that target. If a target is just selected once (instead of double clicked) the robot will simulate a linear or a joint movement from its current position to that target.

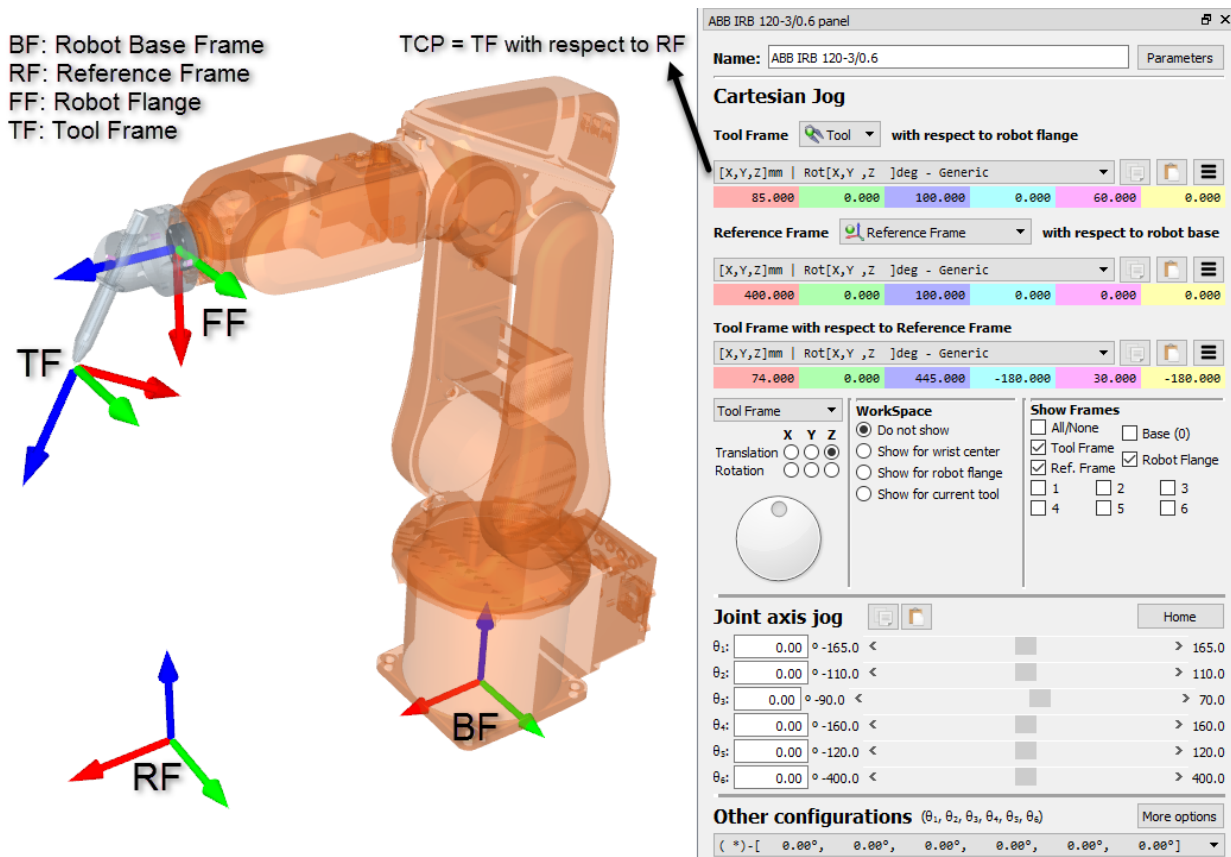


**Tip:** Hold Ctrl down while moving the mouse wheel to change the size of the station tree. Set Tools→Options→Set Default Settings to revert to the default size.


## Robot Panel

Double click a robot to open the robot panel (you can double click it in the tree or the 3D view). It is possible to jog the robot axes using the **Joint axis jog** section and enter specific joint axis values in the text boxes. The joint values and the robot coordinates should match with the values displayed by your robot controller.


You can double click the joint limits to modify the robot axis limits. By default, RoboDK uses the same joint limits used by the robot controller (physical hardware limits). Some applications may require more constrained joint limits (software limits). The joint values can be copied  or pasted  as a list of values, using the corresponding buttons in that section.



The Cartesian Jog section displays all the information related to the robot kinematics:

- The Tool Frame (TF) with respect to the Robot Flange (FF) defines where the selected Tool Frame is located with respect to the Robot Flange. The Robot Flange is always the same, however, the Tool Frame changes depending on the tool that is mounted on the robot. This relationship is also known as UTOOL, ToolData or just Tool in most robot controllers. The Robot Tool is also known as the TCP (Tool Center Point). The Selected Tool becomes the “Active” tool. The active tool is used when creating new targets and programs. The selected tool displays a green mark in its icon: .

**Note:** Modifying the Tool Frame in the robot panel will modify the position of the selected/active tool. This has the same effect as modifying the tool from the [Robot Tool](#) settings.

- The Reference Frame (RF) with respect to the Robot Base (BF) defines where the Reference frame is located with respect to Robot Base Frame. The Robot Base Frame never moves, however, different Reference Frames can be used to position any objects with respect to the same Robot Base Frame. This relationship is also known as UFRAME, WorkObject MFRAME or Reference in most robot controllers. The selected reference frame in the robot panel becomes the “Active” reference frame. The active reference frame is used as a reference for new targets and robot programs. The selected reference frame displays a green mark in its icon: .

**Note:** Modifying the Reference Frame in the robot panel will modify the position of the selected/active Reference Frame. This has the same effect as modifying the Reference Frame from the [Reference Frame](#) settings.

- The Tool Frame (TF) with respect to the Reference Frame (RF) shows the position of the active TCP with respect to the active reference frame for the current position of the robot. Modify this value to move the robot. The joint axes are recalculated automatically. These Cartesian coordinates are recorded when a new target is created (Program→Teach Target), together with the robot axes. The target is also attached to the Active reference frame.


**Tip:** Using the mouse wheel on top of each colored case with the coordinates will update the value and move the robot.

A list of possible configurations is available in the **Other configurations** section. The robot configuration defines a specific state of the robot without crossing any singularities. Changing the configuration requires crossing a singularity. More information available in the [Robot Configurations](#) section.

Finally, the **Parameters** button at the top right allows making some kinematic adjustments, selecting the preferred post processor or extract accurate parameters after a robot calibration project. Modifying these values should only be required in specific circumstances.

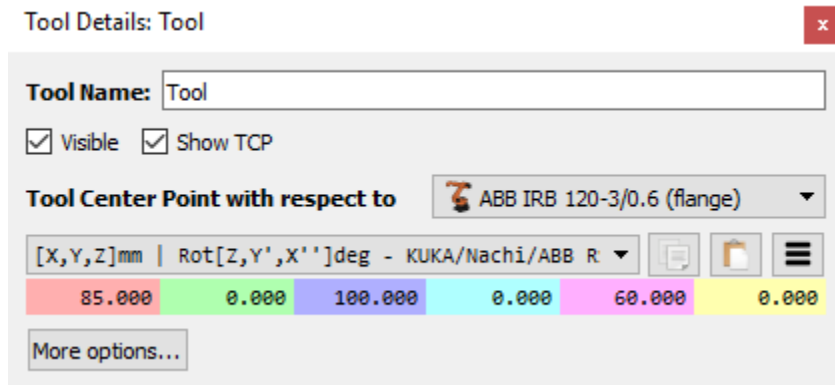
**Video:** the following video provides a general overview of robots in RoboDK and the robot panel: [https://www.youtube.com/watch?v=z357z1NlkZM&list=PLjiA6TvRACQd8pL0EnE9Djc\\_SCH7wxxXI&index=8](https://www.youtube.com/watch?v=z357z1NlkZM&list=PLjiA6TvRACQd8pL0EnE9Djc_SCH7wxxXI&index=8)

## Robot Tool (TCP)

Double click a Robot Tool  to see more details about that tool and modify the position of the Tool Frame with respect to the robot flange, also known as Tool Center Point (or TCP).

**Note:** The TCP pose can also be modified from the [Robot Panel](#).

Selecting **More options...** allows applying a scale factor to the geometry of the tool or move the geometry with respect to the robot flange. Altering these values has no impact on robot programs. The geometry is used for display purposes and collision checking. (keeping the TCP intact).




Tool Details: Tool					
Tool Name: Tool					
<input checked="" type="checkbox"/> Visible		<input checked="" type="checkbox"/> Show TCP			
Tool Center Point with respect to		ABB IRB 120-3/0.6 (flange)			
[X,Y,Z]mm   Rot[Z,Y',X'']deg - KUKA/Nachi/ABB R					
85.000	0.000	100.000	0.000	60.000	0.000
More options...					

**Tip:** It is possible to provide the position of the TCP with respect to another TCP by selecting that reference TCP instead of the robot flange. For example, a cutter tool can be defined at a specific distance along the Z axis with respect to another TCP that represents the spindle. [This example](#) provides more details.

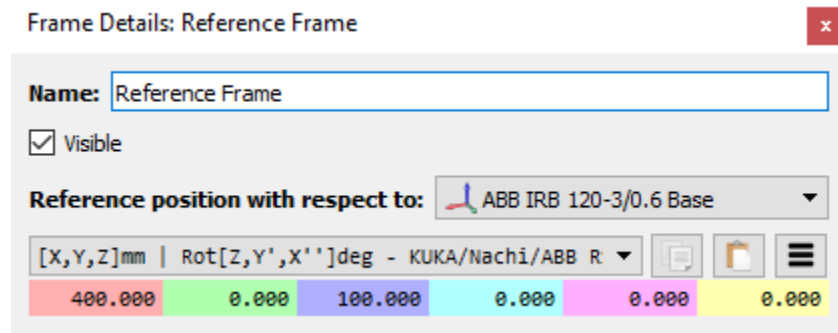
**Video:** The following video provides a general overview of robot tools (TCP): [https://www.youtube.com/watch?v=FltOFBCzx-w&list=PLjiA6TvRACQd8pL0EnE9Djc\\_SCH7wxxXI&index=11](https://www.youtube.com/watch?v=FltOFBCzx-w&list=PLjiA6TvRACQd8pL0EnE9Djc_SCH7wxxXI&index=11)

## Reference Frame

Double click a Reference Frame  to see more details about that reference frame and modify its position with respect to the robot base frame or any other reference frame available in the station. By default, the

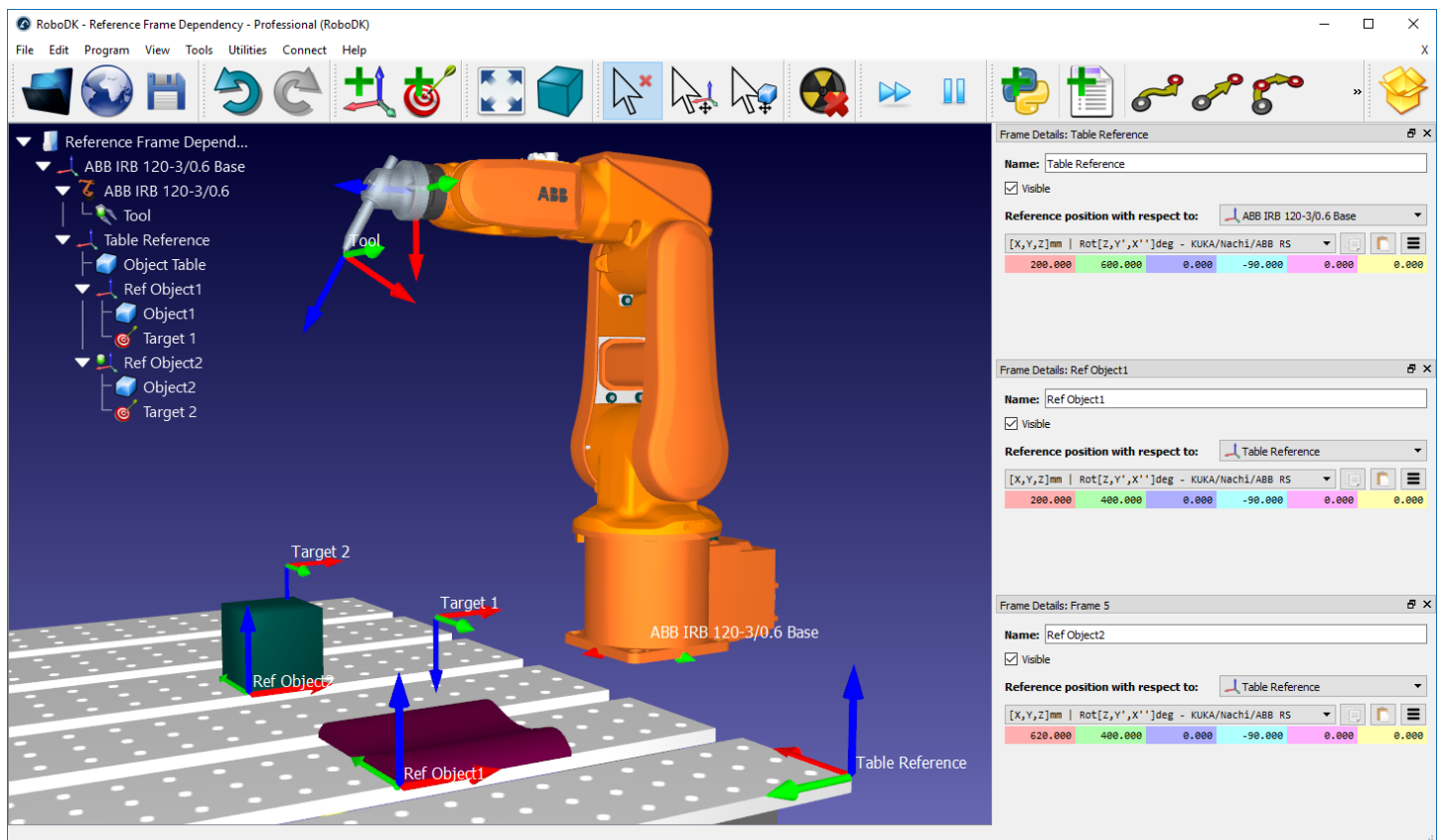
coordinates displayed (pose) are relative to the parent of the reference frame (in this example, the robot Base Frame).

**Note:** The Reference Frame can also be modified from the [Robot Panel](#).



Multiple reference frames can be related to each other to build the dependency that exists in a real application. For example, a table can have a specific position with respect to the robot. Then, two or more objects in the table can have a specific position with respect to the table reference. Moving the table reference will not alter the relationship between the objects and the table but will alter the relationship of all the objects with respect to the robot. The following image shows such an example.

**Tip:** It is recommended to move objects using reference frames. In this case, it is easier to move them in the station and use the same reference frame as a reference for the robot targets.



**Video:** Learn more about reference frames and why they are important for simulation and offline programming: [https://www.youtube.com/watch?v=GGc\\_bXP52dk](https://www.youtube.com/watch?v=GGc_bXP52dk).



## Robot Target

Robot targets allow you to record specific robot positions so that the robot can be moved to that location.

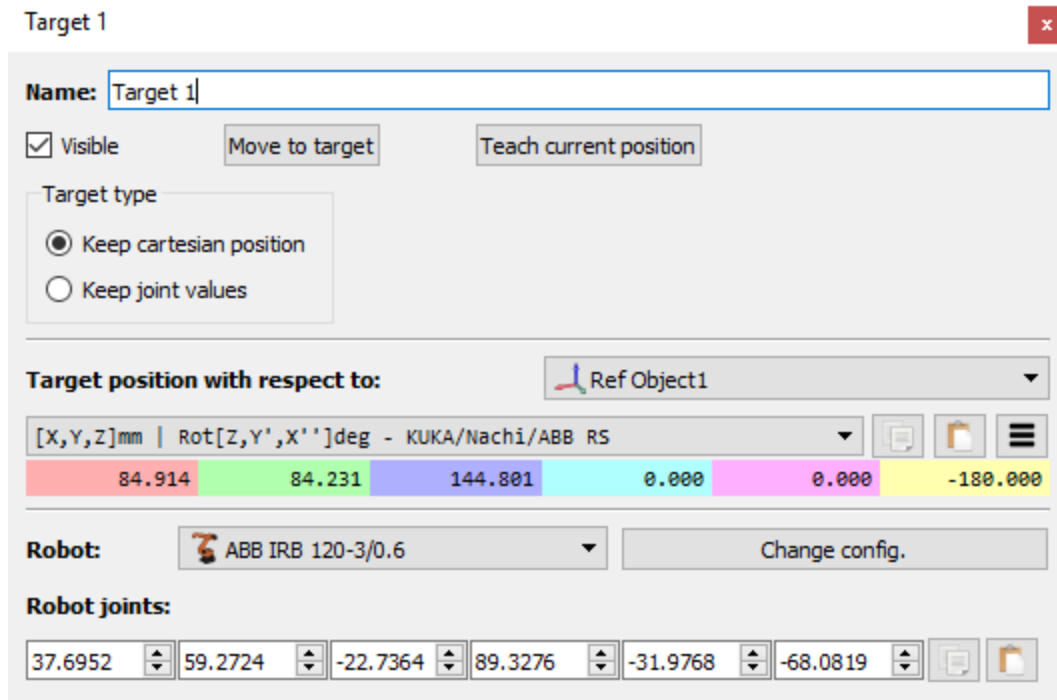
Follow these steps to add a new target and see the information attached to it:

### RoboDK Interface

- Select **Program** →  **Teach Target (Ctrl+T)** to create a new target.

This will record the current position of the robot using the active reference frame  and the active tool frame . The target will be added to the active reference frame.

- Right click a target, then select **More Options... (F3)** to see the recorded pose and joint values.



Target 1

Name: Target 1

☒ Visible    Move to target    Teach current position

Target type

☒ Keep cartesian position  
☐ Keep joint values

Target position with respect to: Ref Object1


[X,Y,Z]mm | Rot[Z,Y',X'']deg - KUKA/Nachi/ABB RS


84.914	84.231	144.801	0.000	0.000	-180.000
--------	--------	---------	-------	-------	----------

Robot: ABB IRB 120-3/0.6    Change config.

Robot joints:

37.6952	59.2724	-22.7364	89.3276	-31.9768	-68.0819
---------	---------	----------	---------	----------	----------

Creating a new target records the TCP with respect to the reference frame in the Cartesian space as well as the current robot axes. By default, RoboDK creates targets as Cartesian targets (**Keep cartesian position** ). In this case, if the reference frame is moved, the robot will try to reach the position of the target with respect to that reference frame.

On the other hand, it is possible to specify the target in the joint space (**Keep joint values** ). In this case, the target is an absolute robot position and it will not be altered if reference frames are moved.

It is common practice to use joint targets to reach a first approach position close to the working area, then, Cartesian targets ensure that the toolpath is not altered if the reference frame or the tool frames are modified.

**Tip:** Selecting **Move to target** will have the same effect as selecting that target in the tree or the 3D view.

It is possible to see other configurations to reach the same pose with the robot. More information in the [next section](#).

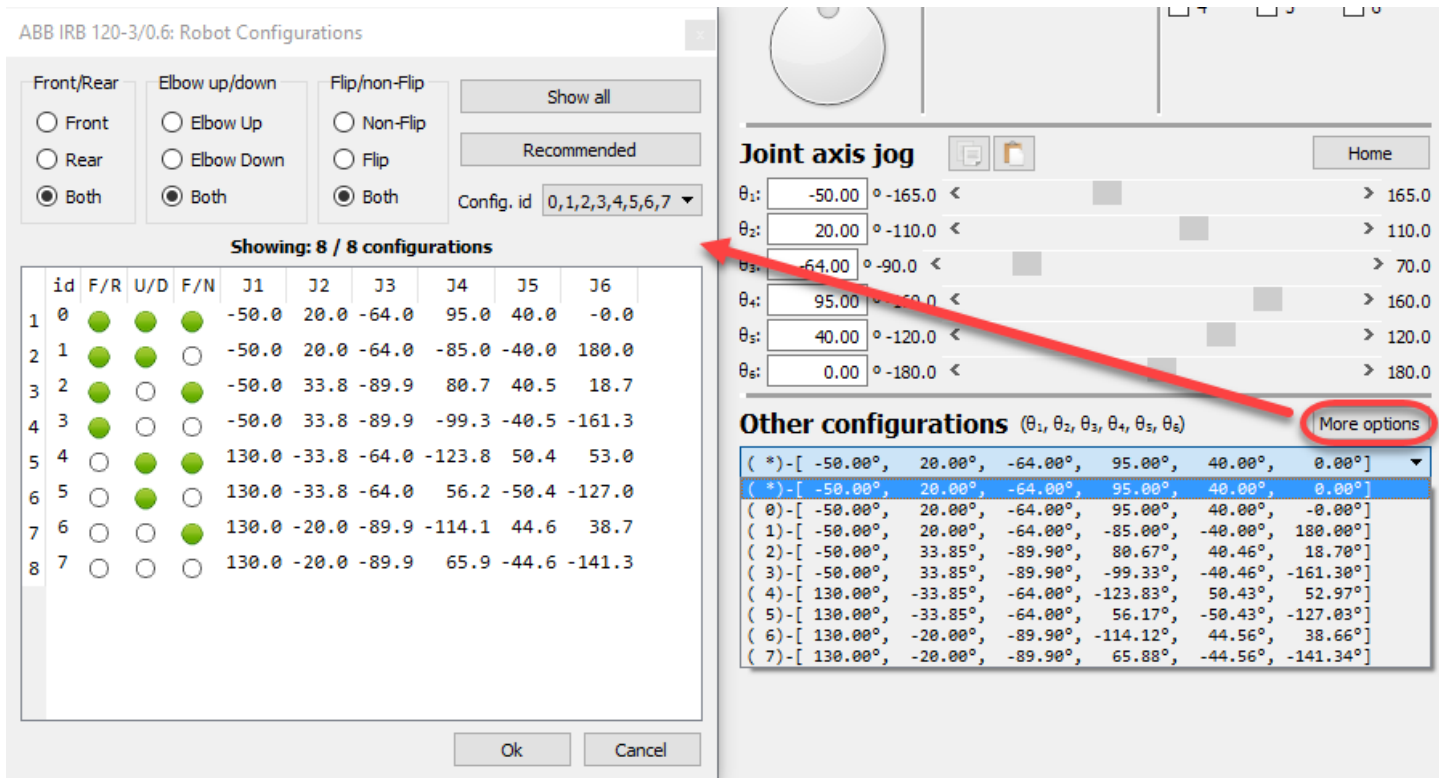
## Robot Configurations

One robot configuration defines a specific state of the robot. Changing the configuration requires crossing a singularity. Robot controllers can't cross a singularity when a linear movement is being made (a joint movement would be required for that).

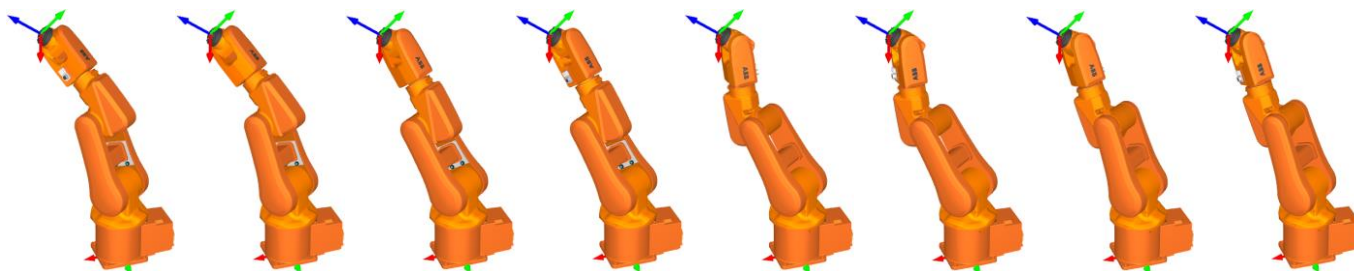
In other words, to accomplish a linear movement between two targets the robot configuration must be the same for the complete movement, including the first and last points.

Right click a robot and select **Change configuration** to open the robot configurations window. It is also possible to open this window by selecting **More options** in the robot panel.





For a standard 6-axis robot there are typically 8 different configurations for any position of the robot if we assume each robot axis can move one full turn. In practice, joint limits can be more or less constrained depending on the robot. Therefore, it may be possible to have from 1 to more than 100 different robot configurations for a specific location depending on the robot.

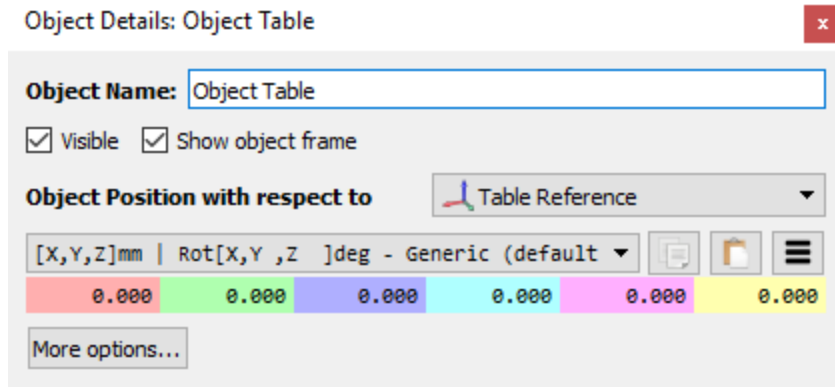


One robot configuration defines a specific way (assembly mode) of reaching a position with the robot. For example, the robot can have the elbow up or the elbow down (Up vs. Down, or U/D), at the same time it can be facing the target or the base can rotate 180 degrees to reach the target backwards (Front vs. Rear, or F/R). Finally, joint 5 can flip by switching the sign at the same time axis 4 and axis 6 compensate for that move (Flip vs. Non-Flip, or F/N). In total, this provides the  $2 \times 2 \times 2 = 8$  configurations.

## Object Settings

Objects can be loaded in RoboDK using 3D file formats such as STL, STEP or IGES. Double click an object in the tree or the 3D view to open the settings window.





It is possible to set or view the position with respect to any reference frame. However, objects are usually set up on reference frames and it is recommended to move the reference frame if an object needs to be moved. Objects can also be grabbed by robot tools after a certain simulation event occurs.

**Tip:** If you want to move an object to a specific location it is better to place the object in a reference frame and move the reference frame instead. This will help defining the location of the object with respect to the robot if you are planning to use an object to create a new robot program (for example, using the curve follow tools or the teach target on surface tools). You can drag and drop an object to a reference frame within the tree to specify this dependency.

The **More options...** menu allows changing the object color, applying a scale factor or moving the geometry with respect to its own reference frame.

**Tip:** It is possible to merge multiple objects in one by selecting two or more objects (holding Ctrl key), then, right click and select **Merge**.


## Main Menu

The Main menu is divided in the following sections:


1. [File menu:](#) Allows importing new files (3D geometry, robots, tools, toolpaths, ...) and opening or saving RoboDK projects (RDK file extension).
2. [Edit menu:](#) Allows to cut/copy/paste an item or a group of items and do undo/redo actions.
3. [Program menu:](#) Allows creating or modifying robot programs and other related options for Offline Programming (OLP).
4. [View menu:](#) Provides useful actions to navigate in 3D and set up specific views.
5. [Tools menu:](#) Provides general tools such as checking for collisions, measuring points, or opening the main options.
6. [Utilities menu:](#) Allows performing specific operations such as using robots for manufacturing operations, calibrating a TCP or a reference frame, using robots as a 3D printer or as a 5-axis CNC, calibrate a robot... These operations might require a specific license option.
7. [Connect menu:](#) Allows connecting to a robot, a measurement system or simulate cameras.
8. [Help menu:](#) Allows opening the online documentation (F1), check for updates or set up a license.


## File Menu

It is possible to Open, Save or export documents from the File menu.


 **New Station** will add a new station in the tree. One station can be loaded or saved as one RDK file. The RDK file (RDK extension) holds all the information about the robots and objects so it is not required to keep a separate copy of the imported items.


**Tip:** It is possible to have more than one Station open at the same time. Double clicking the station icon in the tree will display that station.

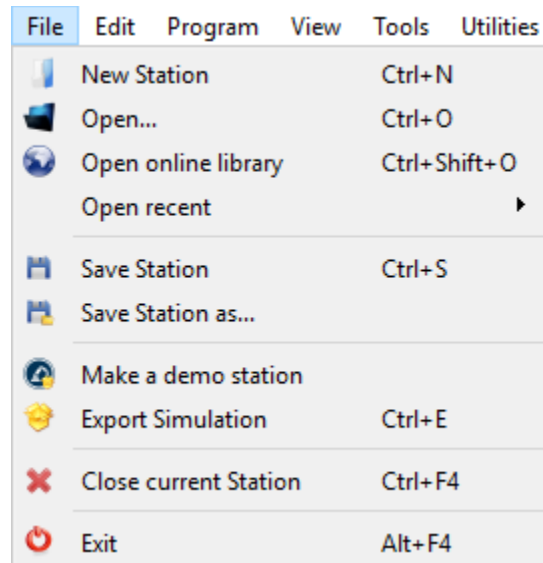
 **Open** will load a new RoboDK file (RDK Station) or import any other recognized file formats, such as .robot for robot files, STEP/IGES/STL for objects, .tool for tool files, etc.

 **Open online library** will show a new window with the library available online.

 **Save Station** will save the RDK file. Select  **Save Station as...** to provide the file location.




 **Make a demo** station will export the station as an EXE file with a simplified version of RoboDK.

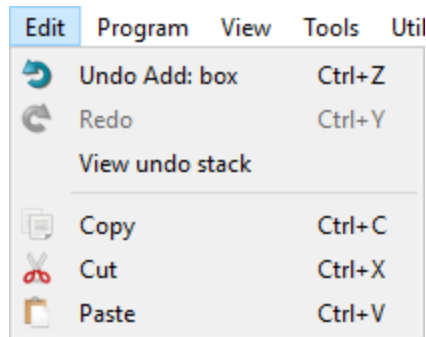
 **Export Simulation** will export a specific program or simulation as a 3D PDF or 3D HTML file. [Example](#).



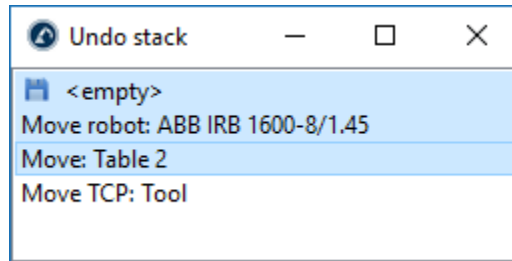
## Edit Menu

Undo (Ctrl+Z) and Redo (Ctrl+Y) actions are accessible from the Edit menu. The history of undo actions is also available and allows reverting changes, backwards or forward, to a specific state by selecting the action.

It is also possible to  cut (Ctrl+X),  copy (Ctrl+C) or  paste (Ctrl+V) one item or a group of items from the station tree. If an item is copied, all the items attached to it are also copied.



**Tip:** It is possible to group similar actions in one. For example, if the robot is moved manually, every small step is recorded (by default). To group all the small steps in one: select Tools→Options→check Bundle similar actions together.



### Program Menu

The program menu contains all the components related to Offline Programming (OLP) and program generation. It is possible to add new programs, reference frames, targets or tools to robots. These Offline Programming components (reference frames, tools, targets, etc.) appear on all programs generated offline.



**Add Reference Frame** will add a new reference frame attached to the station root or attached to another reference frame if that reference frame was selected.



**Add empty tool** will add a new TCP to a robot. No geometry is required to add a new tool. Multiple Tools allow referencing different parts of the same geometry linked to one tool.



**Teach Target (Ctrl+T)** will add a new target to the Active reference frame for the Active robot tool. The active reference frame and active tool can be selected in the [robot panel](#). It is also possible to right click a reference frame or a tool to make them active.



**Teach Targets on Surface (Ctrl+Shift+T)** will allow the user to select points of an object to easily create targets. An example is available in [this section](#).




**Add Program** will add a new program that can be created using the RoboDK Graphical User Interface (GUI). No programming experience is required to create or modify this type of robot program. The robot program can be simulated and generated for a specific robot, automatically and easily.

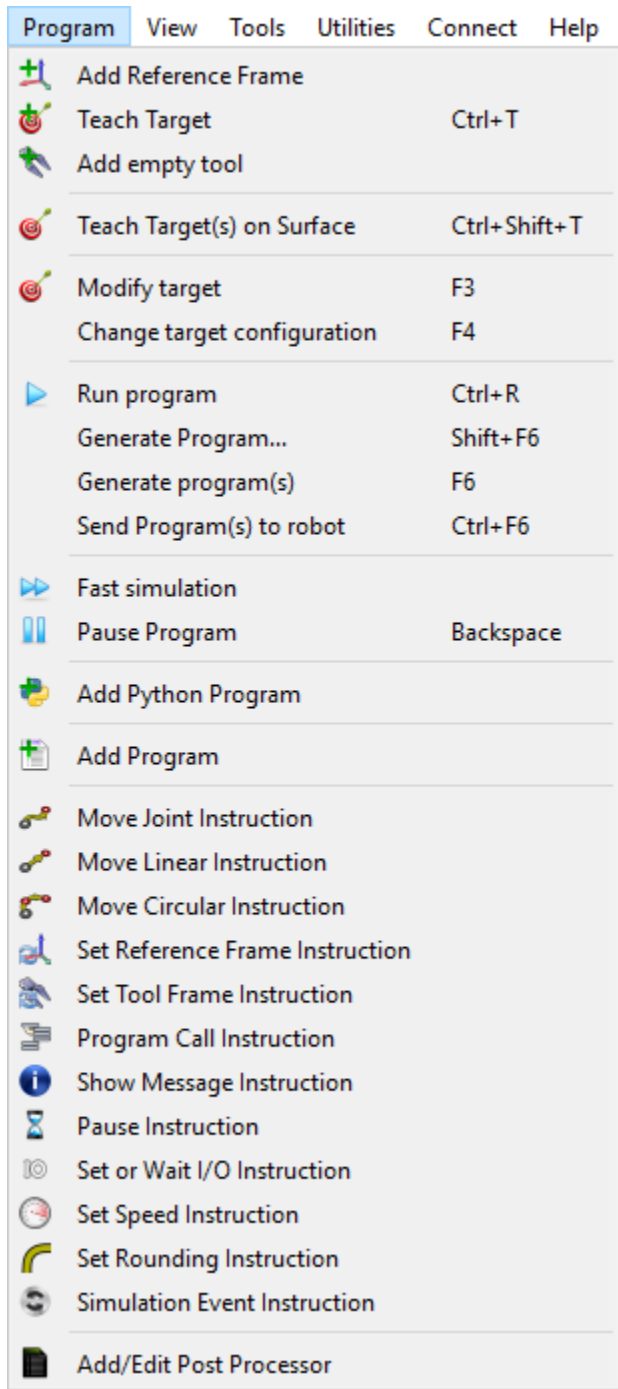
The [Program Instructions](#) section of the Offline Programming document provides more information about available program instructions through the GUI.



**Add Python Program** option will include a sample Python program/macro/script/module in the station that links to the RoboDK API. A Python program using the RoboDK API allows creating robot programs from generic programming code (Python). It is possible to deploy these programs for any specific robot controller. It is also possible to simulate specific tasks to extend the GUI programs. These tasks can be robot subprograms for offline programming, online programming or simply simulate specific events such as making objects appear automatically at random spots for a pick and place simulations. A Python program is like a text file embedded in the station and contains Python code to automate specific tasks in RoboDK. The RoboDK API is deployed using Python by default but other programming languages can be used to interface with RoboDK.

**Note:** The difference between a normal program (using **Add Program**) generated using the Graphical User Interface (GUI) and a Python program (using **Add Python Program**) is that the first does not require programming experience. On the other hand, a Python macro using the RoboDK API allows extending the limits of a program generated using the graphical user interface.

Finally, it is possible to  **Add or Edit Post Processors**. Post Processors define the way programs are generated for a specific robot controller, allowing to accommodate vendor-specific syntax. Post Processors are final component of the offline Programming Process.



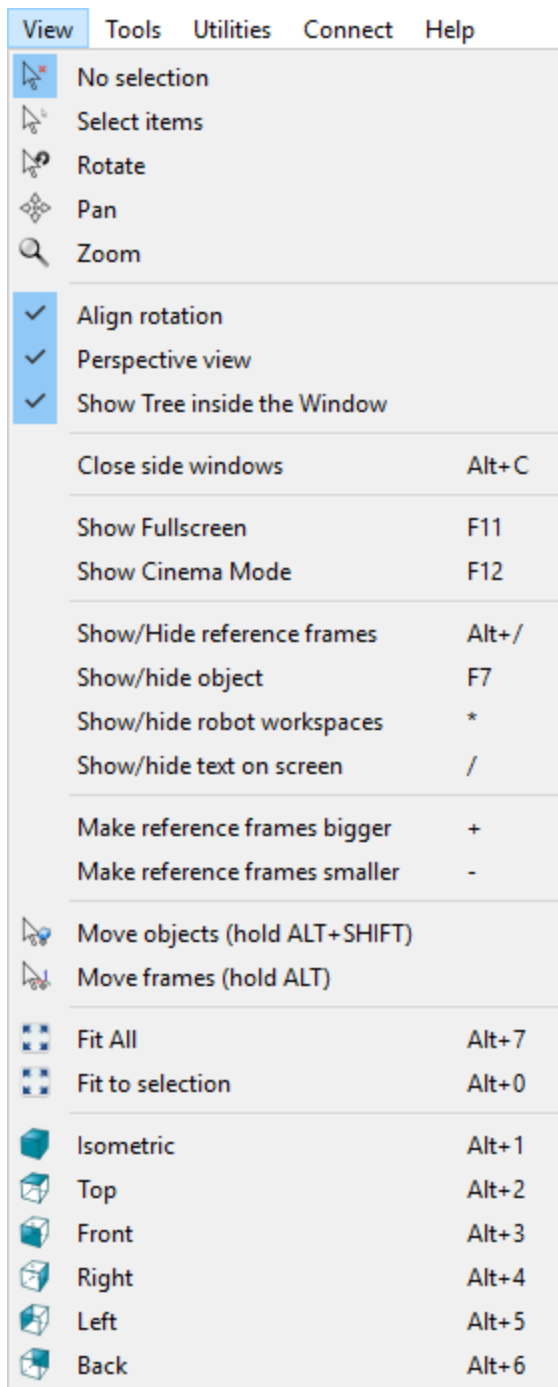
### View Menu

Most options required to navigate in 3D are available from the View menu. It is possible to Rotate, Pan and Zoom from this menu (as well as by right clicking the 3D view). This is useful for navigating in 3D using a laptop touchpad (instead of a mouse).

To allow a free rotation in any direction uncheck the option: View→Align rotation. Otherwise, RoboDK locks the station reference to keep the XY plane horizontal by default.

It is possible to show or hide the robot workspace by selecting the asterisk key (\*). It is also possible to switch between visible and invisible items by selecting the F7 key.



**Tip:** It is possible to make the reference frames bigger or smaller by pressing the + or – key multiple times. If a lot of items are visible this is useful to adjust the size of the reference frames and properly grab them if they need to be moved from the 3D view (by holding the ALT key for example).




## Tools Menu


Generic tools are available in the Tools menu, such as taking snapshots of the 3D view, activating the robot trace, activate collision checking or measuring point coordinates.

Activating the **Trace** will show the trace of all robots as they move.

 **Check collisions** will activate or deactivate collision checking. When collision checking is activated, objects that are in a collision state will be displayed in red.  The **Collision map** allows specifying what object interactions are being checked.


**Note:** More information available regarding collision detection in the [Collisions](#) section.

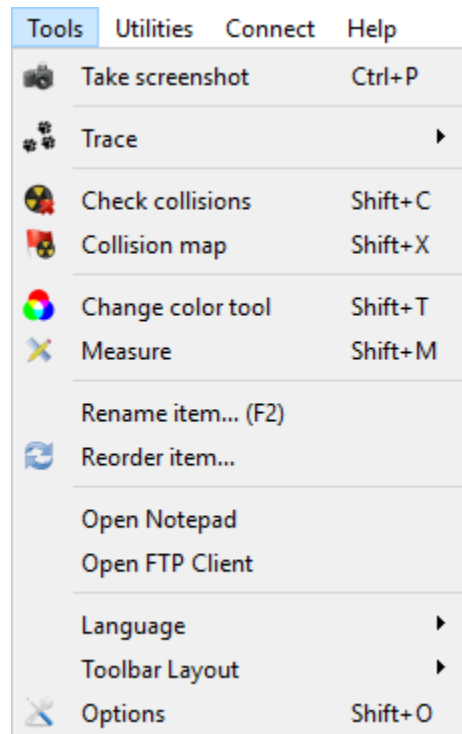
 **Change color tool** will display a small window that allows changing the color of robots and objects. It is also possible to flip the normal vectors of surfaces.

 **Measure** will display a window that allows measuring points in 3D with respect to a local reference frame or the station reference frame (absolute measurements).

It is possible to specify the language of the RoboDK application by selecting **Tools→Language** and select the preferred language. RoboDK will be displayed in the selected language immediately.


**Toolbar Layout** allows setting up the default toolbar. Alternatively, it is possible to specify a toolbar for a more basic or more advanced usage.


Select  **Options** to open the main options menu. More information available in the [Options Menu](#) section.





### Utilities Menu


The utilities menu allows performing specific tasks:


 **Calibrate Tool frame (TCP)** allows calibrating a robot TCP by providing data from the real setup, such as the joint configurations to reach a point using different orientations. This procedure is usually available from most robot teach pendants. RoboDK allows calibrating a TCP with as many configurations as desired. Using more configurations allows obtaining a more accurate TCP value. [Read more about TCP calibration.](#)

 **Calibrate Reference Frame** allows identifying a reference frame with respect to a robot base frame. This allows accurately matching the part from the real setup to the virtual environment. [Read more about Reference Frame calibration.](#)

 **Synchronize External Axes** allows setting up one or more external axes and a robot as one robot mechanism. More information available in the [External Axes section](#).


 **Robot Milling project** can easily convert machine toolpaths to robot programs. RoboDK can import programs made for 5-axis CNC using CAM software, such as generic G-code or APT files. These programs/toolpaths can be easily simulated and converted to robot programs with RoboDK. More information available in [this](#) section.

The  **Curve follow project** is like a Robot Milling project but it allows selecting curves extracted from 3D geometry as toolpaths. It is also possible to select **Import Curve** to import 3D curves from CSV or TXT files. These curves must be provided as a list of XYZ points and, optionally, IJK vector. More information available in the [curve follow project](#) section.

The  **Point follow project** is like a Robot Milling project but it allows selecting points extracted from 3D geometry and easily create a robot toolpath. It is also possible to select **Import Points** to import 3D points from CSV or TXT files. These points must be provided as a list of XYZ points and, optionally, IJK vector. More information available in the [point follow project](#) section.

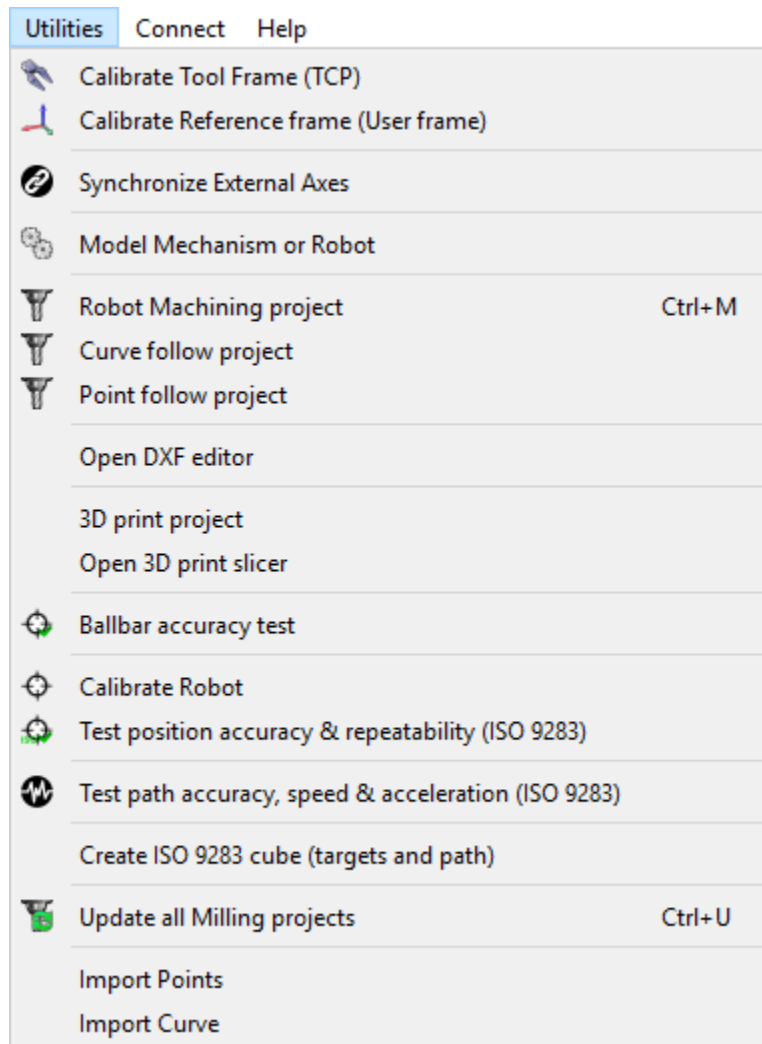
Select **3D print project** to generate a robot 3D printing program for a specific object. The object must be available in the RoboDK station. The 3D printing toolpath is converted to G-code behind the scenes using a Slicer and then treated like a 3-axis machining toolpath. More information available in the [robot 3D printing](#) section.

The Ballbar Accuracy test allows checking the robot performance using a Telescoping Double Ballbar device. More information about robot ballbar testing available here: <https://robodk.com/ballbar-test>.

 **Calibrate Robot** allows setting up a robot calibration project to improve robot accuracy and find robot error parameters. A calibrated robot can be used in any RoboDK Offline Programming project. Robot calibration usually improves robot accuracy by a factor of 5 or better, depending on the robot model. Robot calibration requires using measurement systems to take robot measurements. Robot accuracy and repeatability can be tested with ISO9283 before and/or after calibration. More information about robot calibration and performance testing here: <https://robodk.com/robot-calibration>.

**Note:** Some of these tools might require a specific license that extends the default offline programming license.



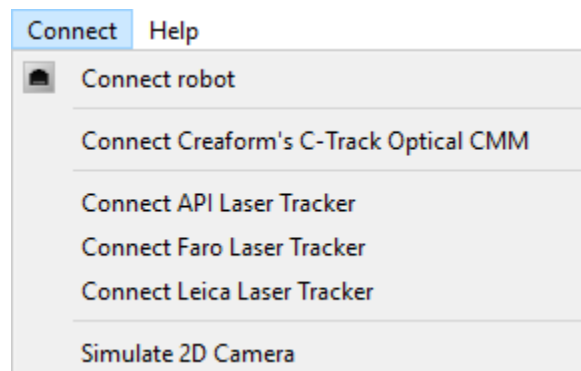


### Connect Menu


It is possible to connect to a robot and enter the connections parameters, such as the robot IP, FTP username and FTP password. Setting up a robot connection allows transferring programs through FTP or running programs directly from the PC.

New robot drivers can be developed by end users, more information available in the [robot drivers section](#).

It is also possible to connect to measurement systems such as laser trackers or the Creaform Optical CMM. This allows to fully automate robot calibration and performance testing.



### Help Menu

 **Help (F1)** opens this documentation online. A PDF version of the documentation is available for download at the top of each section. When you press **F1**, RoboDK displays the Help topic related to the item currently selected.

Select **Check for updates...** to check if an update is available. A message will pop up with a recommended update or just notifying that the current version is already up to date. If no message pops up it means that a firewall is blocking communication between RoboDK and internet.

