

# RoboDK for KUKA Robots

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



# Contents

KUKA robots.....	1
Transfer a robot program .....	1
Start a robot program .....	1
Retrieving the TCP.....	2
Retrieving the robot joints .....	3
Administrator mode.....	3
RoboDK driver for KUKA.....	4

# KUKA robots

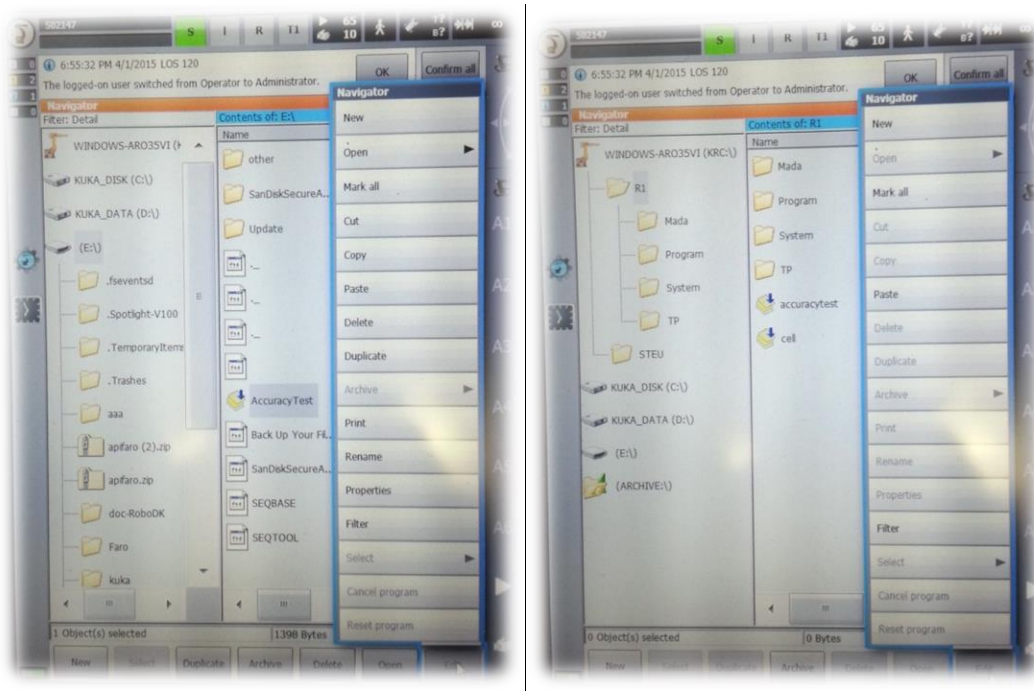
RoboDK supports all KUKA robot controllers since KRC2, including KUKA KRC3 and KRC4 controllers. This documentation is based on a KRC4 controller. The KRC4 robot controller runs the Microsoft Embedded Windows 7 operating system. Previous controllers, such as KRC2, run Windows 95. The robot teach pendant shows an “HMI” which is a program that KUKA developed to run on Windows and it is the interface that the robot user must use to manipulate the robot.

The following sections demonstrate typical operations using a KUKA robot teach pendant to prepare a new program in RoboDK and transfer it to the robot.

## Transfer a robot program

Follow these steps to load a program from a USB disk to your KUKA KRC4 robot controller.

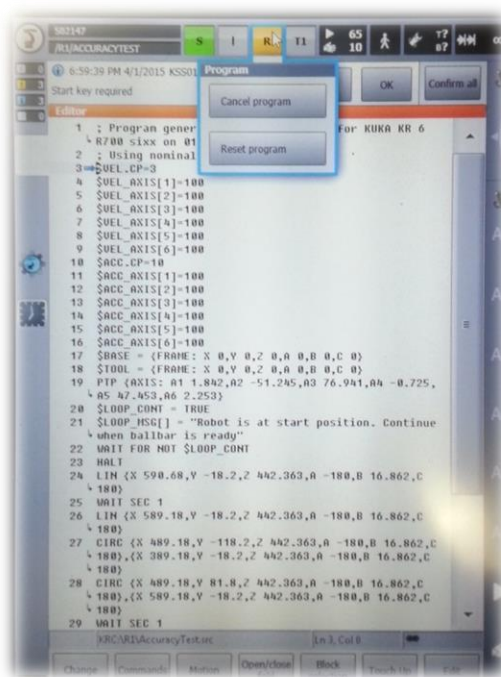
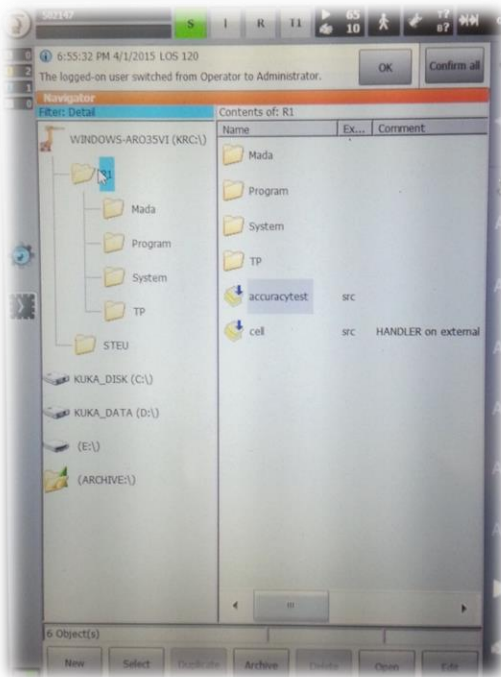
1. Insert the USB disk on the robot controller (it is much faster than using the teach pendant connection)
2. If we don't see the USB disk we must enter in **Administrator** mode
3. Select the file from the USB disk
4. Select **Edit→Copy**
5. Select a folder in the KRC unit
6. Select **Edit→Paste**



## Start a robot program

Follow these steps to start a robot program on your KUKA KRC4 controller.

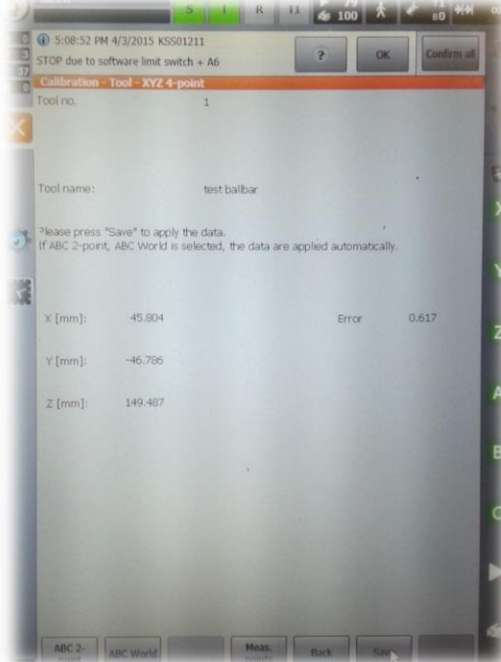
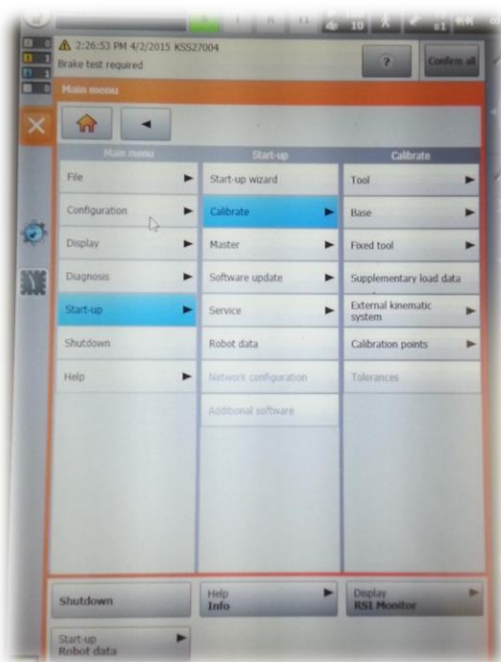
1. Select a program from the KRC memory unit
2. Select **Select** on the screen
3. Select the button “R” (top) and **Reset program**
4. Start the program by selecting green “Play” button on the teach pendant



## Retrieving the TCP

The following steps allow you to create or modify robot tools (TCP) from your robot controller, also known as **\$TOOL** in KUKA KRC robot programming):

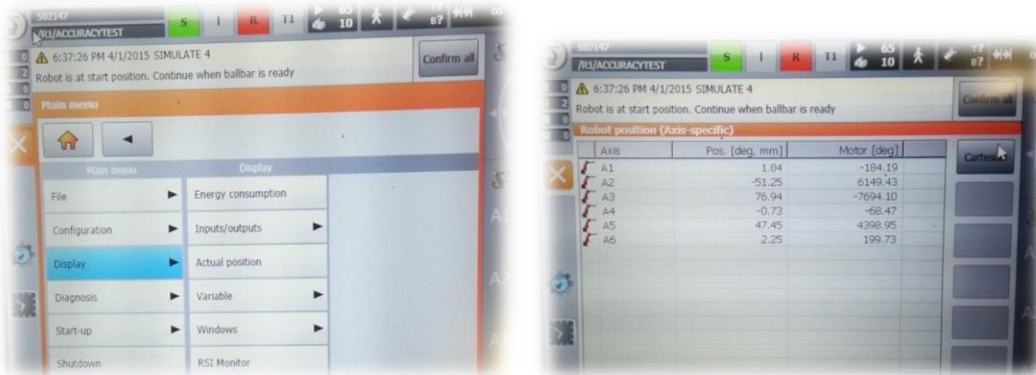
1. Select **MENU→Start-up→Calibrate→Tool**
2. Select a tool and edit or retrieve the X,Y,Z position of the TCP.



## Retrieving the robot joints

The following steps allow you to retrieve the robot joints from your robot:

1. Select **MENU**→**Display**→**Actual position**
2. Select **Joints** mode and use the left column to take the robot joints

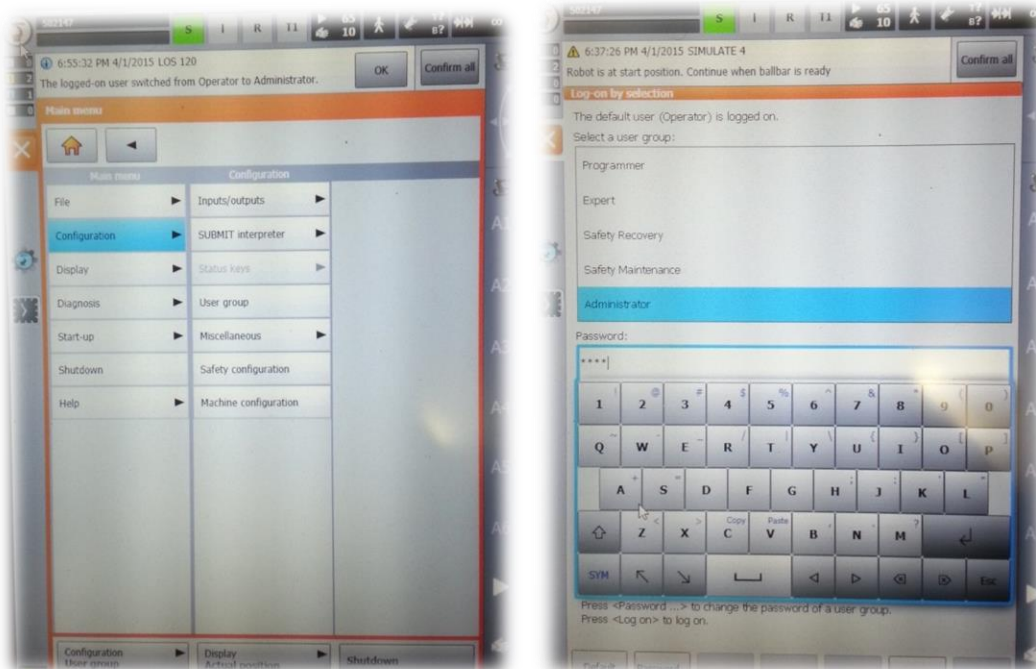


Tip: It is possible to retrieve the robot joints more accurately (5 decimal precision) by monitoring the `$AXIS_ACT` variable or simply using the RoboDK robot driver for KUKA and selecting the **Get robot joints** button

## Administrator mode

Some menu sections require “Administrator” rights. The following steps allow entering in “Administrator” mode:

1. Select **MENU**→**Configuration**→**User group**
2. Select **Administrator** (or other).
3. Enter the password if required (default password is “**kuka**”)







## RoboDK driver for KUKA

Robot drivers provide an alternative to Offline Programming (where a program is generated, then, transferred to the robot and executed). With robot drivers, it is possible to run a simulation directly on the robot (Online Programming). More information available in the [Robot Drivers](#) section.

A connection between RoboDK and the KUKA robot can be established to move the robot automatically from a connected PC using RoboDK. This allows using the RoboDK **Run on robot** option for online programming and debugging. The connection can be established through a standard Ethernet connection (TCP/IP).

Follow these steps to set up the RoboDK driver for KUKA:

1. Connect a mouse (optional, but strongly recommended).  
It is possible to plug USB devices to the teach pendant or the controller (reboot is not required). Alternatively, it is possible to establish a remote desktop connection. These steps can also be accomplished using the teach pendant's touch screen and the virtual keyboard.
2. Using the KUKA HMI application it is possible to open the main menu using the KUKA button , at the top left of the screen:
  - a.  **KUKA→Configuration→User group→** choose **Administrator** (password: **kuka**)
  - b.  **KUKA→Start-up→Service→Minimize HMI** (the windows screen will appear)
3. Copy the KUKAVARPROXY folder on the Desktop (or somewhere in the controller PC)
4. Allow port 7000 for TCP/UDP communication (this step is not required on KUKA KRC2 controllers):
  - a. Select the HMI.
  - b.  **KUKA→Start-up→Network configuration→Advanced**
  - c. **NAT→Add port→Port number 7000**
  - d. Set permitted protocols: **tcp/udp**
5. Start the KUKAVARPROXY.EXE program on the robot controller (running on Windows).

**Note:** Since 2020, new KUKA KRC4 controllers use Windows 10 IOT. If your controller is using Windows 10 you should stop the process `nginx.exe` before starting the KUKAVARPROXY application to allow using the port 7000.

6. The following steps allow starting the driver automatically on the controller on reboot (recommended):
  - a. Create a shortcut of the KUKAVARPROXY.EXE file
  - b. Select Windows **START→All programs→Right click startup→Open**
  - c. Paste the shortcut in the startup folder

The KUKAVARPROXY server is now ready. You can leave this program running. This server allows exchanging global variables from the KUKA controller to the remote PC.

The next steps are to set up the main program that will handle the robot movements:

1. Add the declaration of the following global variables:  
To do so, locate and modify the file "KRC\R1\STEU\config.dat" (or in "KRC\R1\System\config.dat" for KRC2 controllers). The folder "KRC\R1\" can also be accessed from the C:\ drive at the following Windows path: "C:\KRC\ROBOTER\KRC\".

```
INT COM_ACTION=0
INT COM_ACTCNT=0
REAL COM_ROUNDMM=0
REAL COM_VALUE1=0
```

```
REAL COM_VALUE2=0
REAL COM_VALUE3=0
REAL COM_VALUE4=0
DECL E6AXIS COM_E6AXIS
DECL FRAME COM_FRAME
DECL POS COM_POS
```

2. Copy the SRC program provided (**RoboDKsynch.src**) to the folder **KRC\1**.
3. Manually start the **RoboDKsynch.src** program to make the robot behave like a server that responds to move commands coming from the PC.

If the RoboDKsynch.src program is not running, RoboDK will still be able to read the robot joints anytime if the KUKAVARPROXY program is running in the robot controller.