APPENDIX 6

Tutorial on MATLAB and CoppeliaSim Software

1. Download and install CoppeliaSim Edu software at the link below.
   Link: https://www.coppeliarobotics.com/downloads

2. CoppeliaSim are continually updating their software and you might find a problem with the software where it keeps crashing. This is because for some reasons there are disabling the older version of the software. If you encounter this problem, you can try the solution at the link below.
   Link: https://forum.coppeliarobotics.com/viewtopic.php?f=9&t=9334

3. Files required for this tutorial can be downloaded on Google Drive link below. We will go through on each file in this tutorial.
   https://drive.google.com/drive/folders/16WEtHbZDQKhxF__6IJU4rS8ogmjWMNHV?usp=sharing

4. Firstly, open the CoppeliaSim Edu software.

5. Download "*KUKA KR 16.ttm*" from the Google Drive. This is the robotic arm model that we will use for this tutorial. Copy the file and paste it on the directory folder for CoppeliaSim Edu software. Paste it in the "*non-mobile*" folder.

   C:\Program Files\CoppeliaRobotics\CoppeliaSimEdu\models\robots\non-mobile

   After that, based on the image below (Image 13);

   A. Click "*mobile*" first, then click "*non-mobile*" to refresh the folder.
   B. Scroll down until you see the robotic arm model for "*KUKA KR 16*".

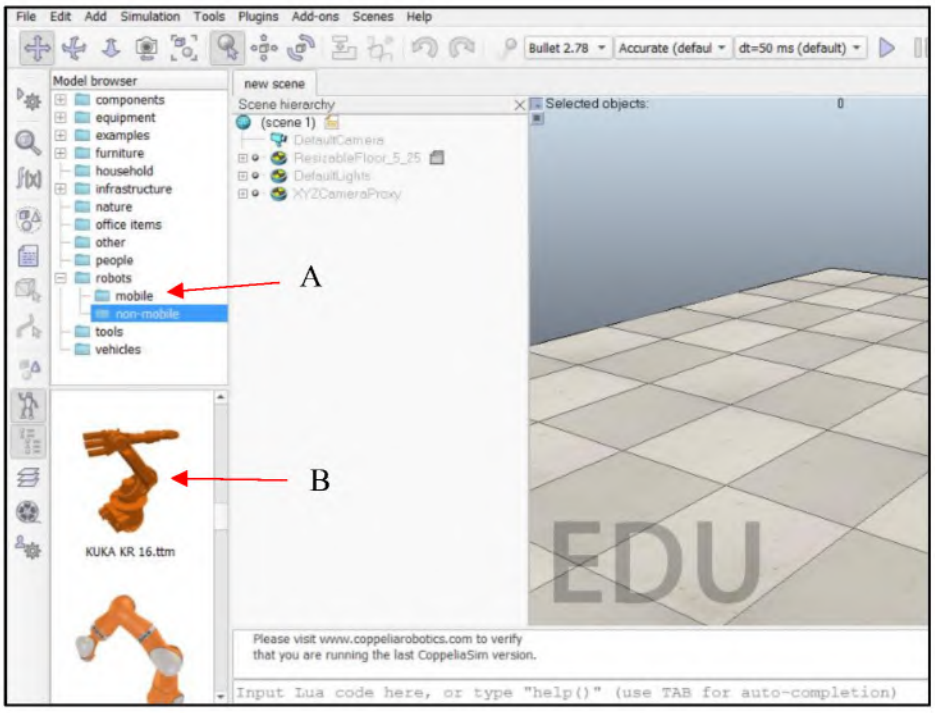


Image 13 CoppeliaSim Edu

6. Left click, hold, and drag the KUKA KR 16 model to the right page that has the square floor. Until the model appear on the page like Image 14.

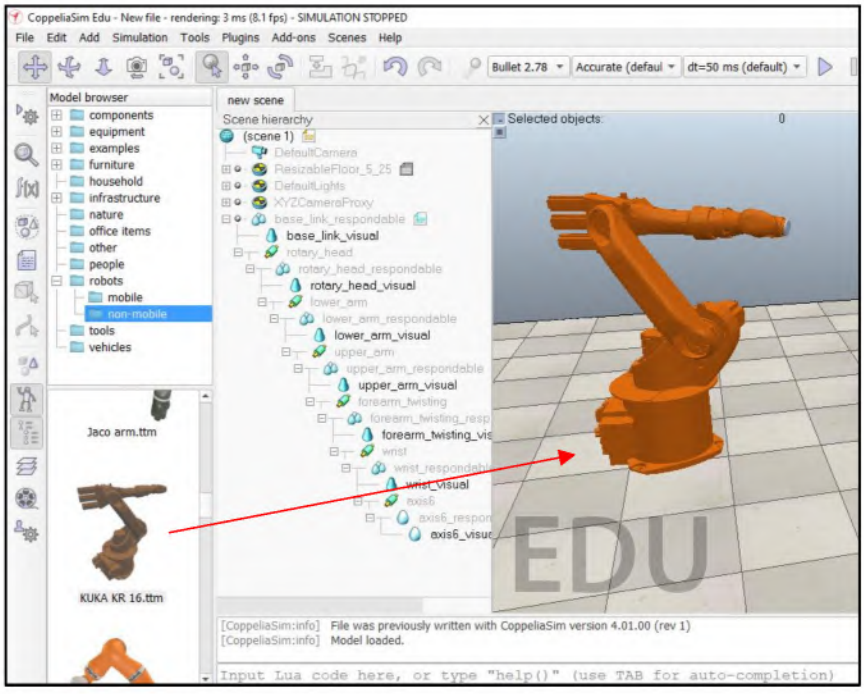


Image 14 KUKA KR 16 Robotic Arm Model

7. Next, based on Image 15, you can change the speed of the robotic arm at (A). The lower the value, the slower the speed of the robotic arm. Then, click "Start Simulation" button at (B) and see what happens to the robotic arm. If any warning window appears, just click "OK".

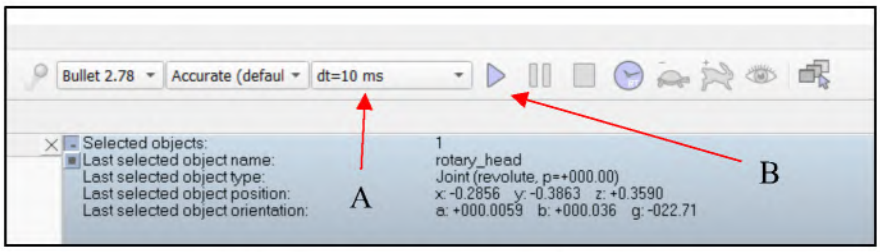


Image 15 Change Speed and Start Simulation

8. Important notes!

   Every robotic arm model, have different settings. For the KUKA KR 16 robotic arm model that you have downloaded from Google Drive, the setting has NOT been set, therefore we will need to edit the setting specifically for this tutorial. The steps for the settings will be explained later. Other than that, you can change the setting of the model for a different application and different objective. You can also refer many tutorials on the internet for the settings that suit the need of your project.

9. After the robotic arm model is successfully imported, we will learn how to communicate CoppeliaSim software with MATLAB. The objective of this task is to control the robotic arm model on CoppeliaSim from MATLAB. Before we start, please take a look of the video, link YouTube below, of the final result for this tutorial. Hopefully, this video helps you to understand the objective of this tutorial.

   YouTube Link: https://www.youtube.com/watch?v=l3b8gnQEw4U

10. The files shown in Google Drive in step 3, are just backup files in case you have a problem with the original files. But, it is recommended to use the original files. You can refer Image 16 and Image 17 to obtained the original files:

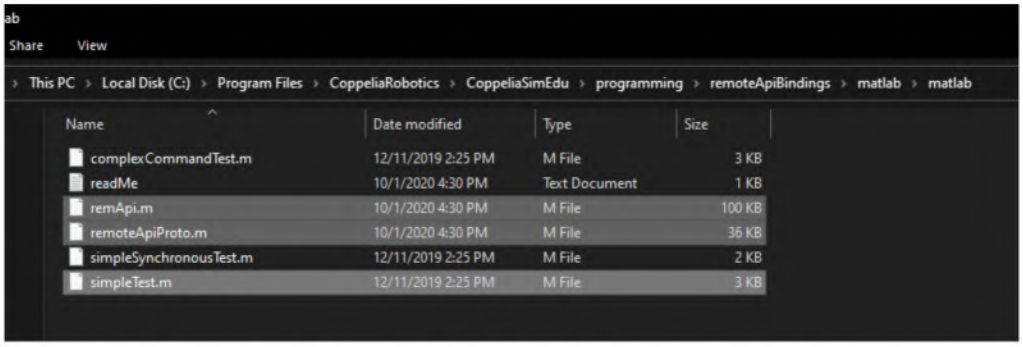


Image 16 C:\Program Files\CoppeliaRobotics\CoppeliaSimEdu\programming\remoteApiBindings\matlab\matlab

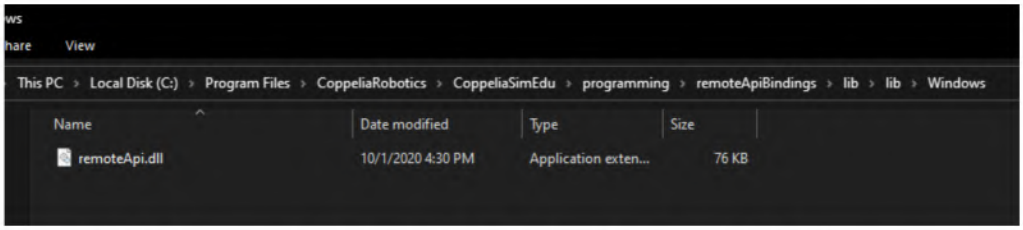


Image 17 C:\Program Files\CoppeliaRobotics\CoppeliaSimEdu\programming\remoteApiBindings\lib\lib\Windows

Next, create a new folder inside the MATLAB folder in the Document, and copy all the required files inside the new folder. Refer image below as an example and make sure all the files are in the same folder.

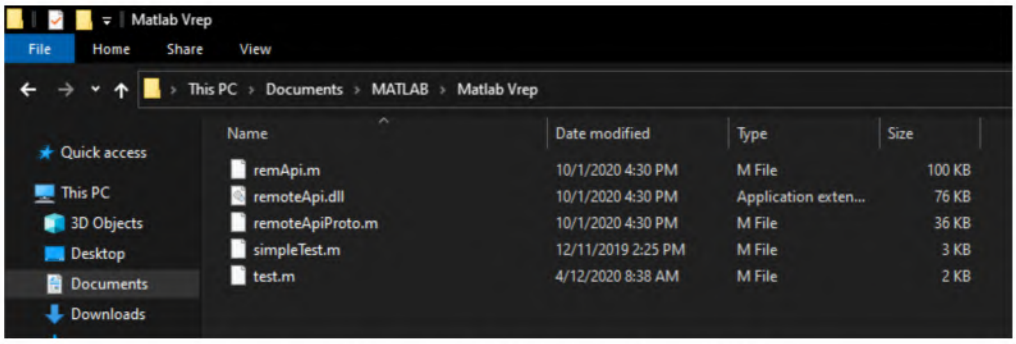


Image 18 Required Files in the Same Folder

11. Next step, copy this line “simRemoteApi.start(19999)” and refer image below on where to paste the line in CoppeliaSim software. A: Double click this icon. B: Paste here. Close the window after paste the line.

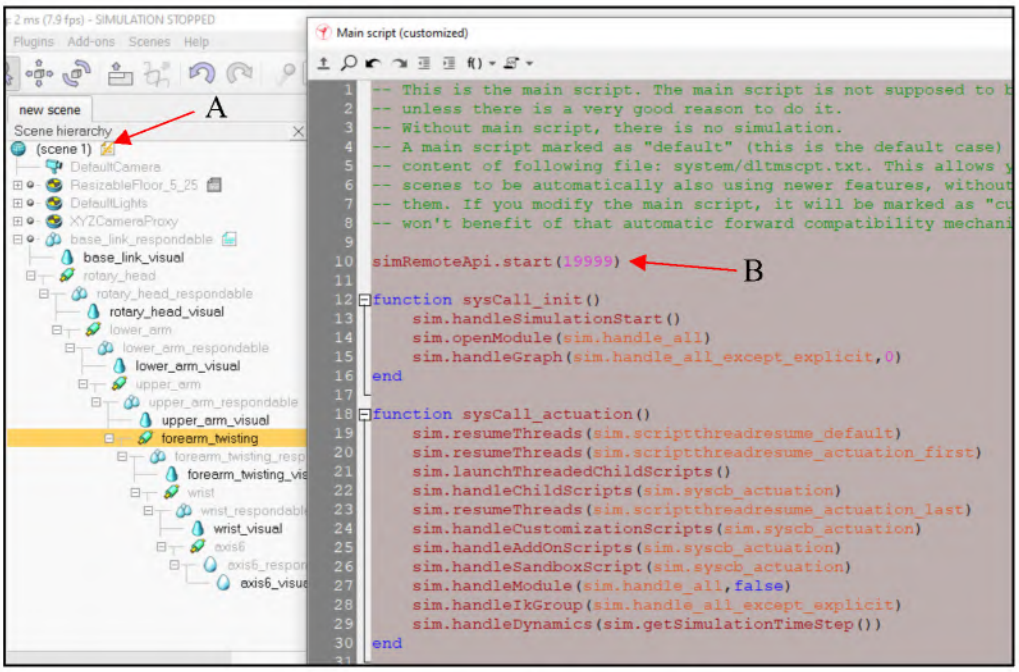


Image 19 Copy and Paste the Line

12. For the next step, please read carefully on how to communicate between the software. Open "*test.m*" in MATLAB software. Then,
    - Click "Run Simulation" on CoppeliaSim software.
    - Click "Run Test" on MATLAB software.

    Take a look at the output shown in the command window on MATLAB software and if the output is the same as in the image below, then you have successfully communicated between the two software.

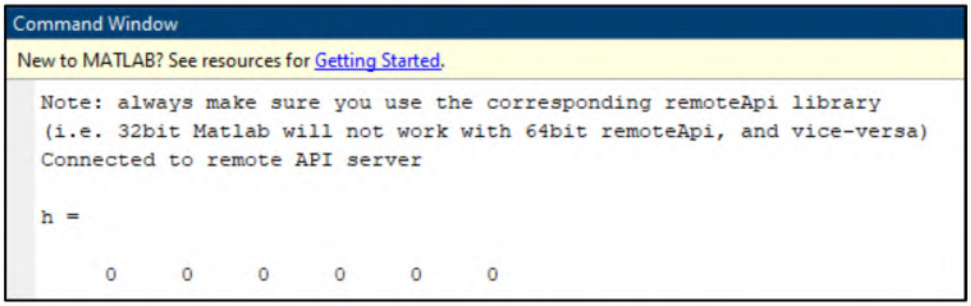


Image 20 Successful Connection

Important information:

The file test.m is created from two sources. The sources are:

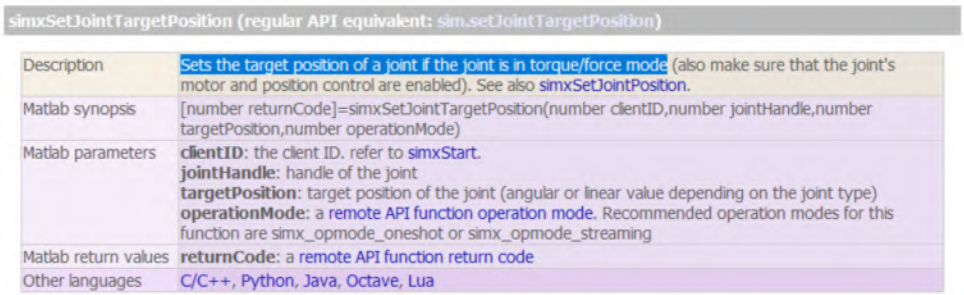
1. File "simpleTest.m"
2. MATLAB remoteAPI function website: https://www.coppeliarobotics.com/helpFiles/en/remoteApiFunctionsMatlab.htm#simxSetJointPosition

Information taken from No. 1, are for the connection, and for No. 2 are the functions to control the robotic arm model on CoppeliaSim. Next, from the file test.m, lines like "sim.simx_opmode_blocking" or "sim.simxGetObjectHandle" can be seen in the file. Now, the most important part is the "sim" in the lines. The "sim" is the new version of CoppeliaSim software where the function needs to have this word. A lot of examples or tutorials on the internet that use "vrep" such as "vrep.simx_opmode_blocking" instead of "sim". The "vrep" is the line for a very old version of CoppeliaSim. Therefore, if you find a tutorial that use "vrep", please change it to "sim".

13. At this point, a successful connection between MATLAB and CoppeliaSim is completed. Now, we will try to control the robot model from the MATLAB. As mentioned in step number 8, we will learn how to setting the robotic arm. Buckle up! The settings are very handy, so hopefully you can stay motivated.

A. Joint Setting (🚀)

Quick information; if you open MATLAB remoteAPI function website shown in Step 12, you can find the function as shown in image below. It says that 'Sets the target position of a joint if the joint is in torque/force mode'.



simxSetJointTargetPosition (regular API equivalent: sim.setJointTargetPosition)

Description	Sets the target position of a joint if the joint is in torque/force mode (also make sure that the joint's motor and position control are enabled). See also simxSetJointPosition.
Matlab synopsis	[number returnCode]=simxSetJointTargetPosition(number clientID,number jointHandle,number targetPosition,number operationMode)
Matlab parameters	**clientID**: the client ID. refer to simxStart. **jointHandle**: handle of the joint **targetPosition**: target position of the joint (angular or linear value depending on the joint type) **operationMode**: a remote API function operation mode. Recommended operation modes for this function are simx_opmode_oneshot or simx_opmode_streaming
Matlab return values	**returnCode**: a remote API function return code
Other languages	C/C++, Python, Java, Octave, Lua

Image 21 MATLAB remoteAPI Function Website

Therefore, we will set the mode for all joint in Torque/force mode. Double click the icon (), and setting the joint as shown in Image below. Make sure to set all the joints with these settings.

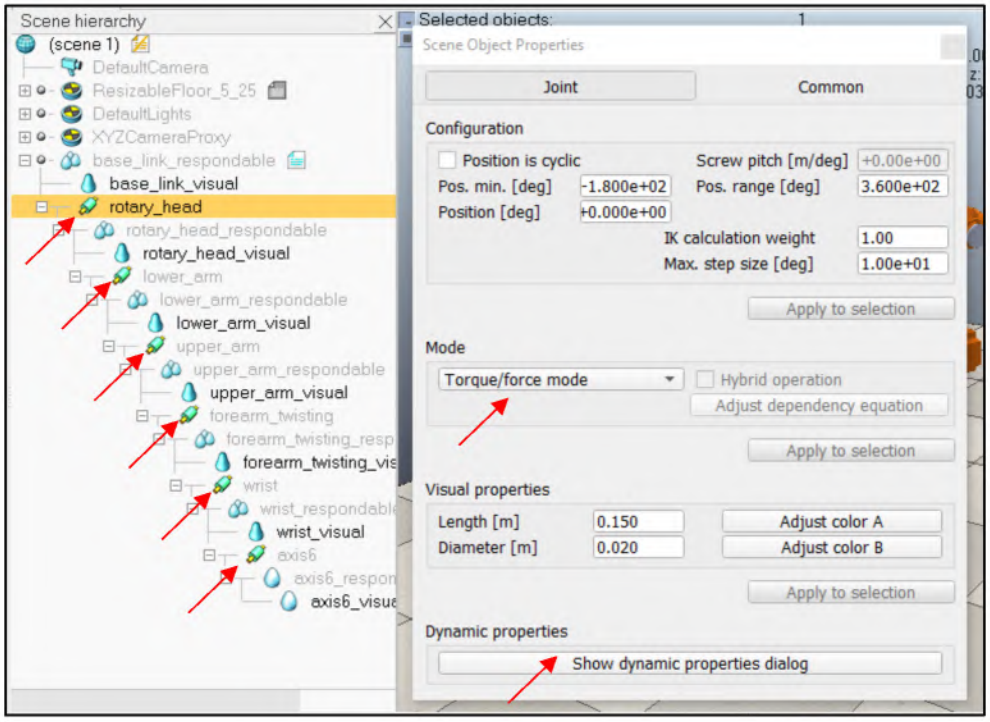


Image 22 Joint Settings

After that, click "*Show dynamic properties dialog*". Attention! Image 23 is only a setting for red arrows only.

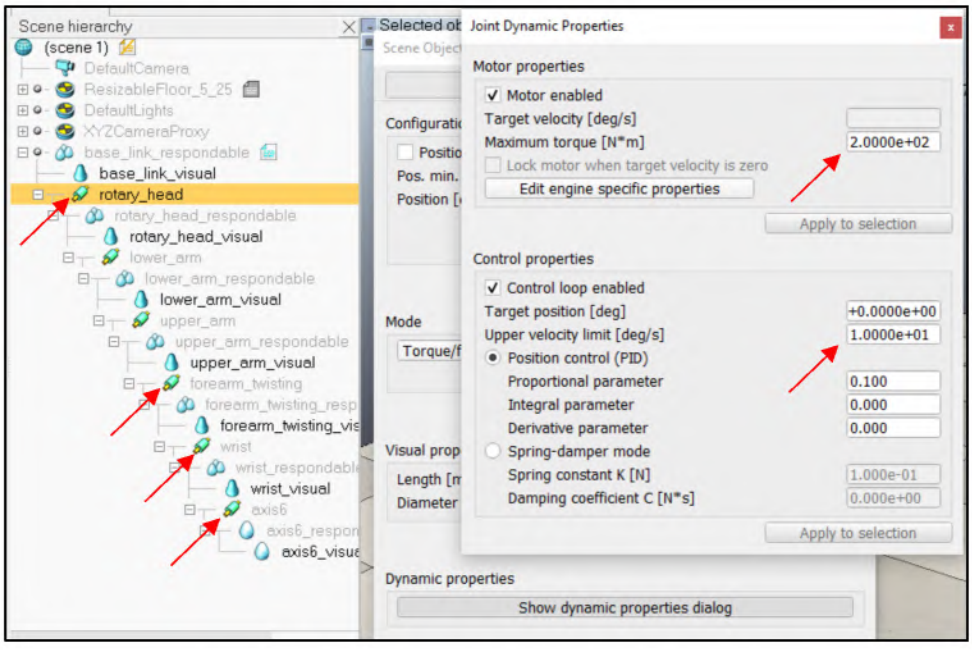


Image 23 Joint Settings for four joints only

As shown in Image 23, the "*Maximum torque [N*m]*" for "*rotary_head*", "wrist", "axis6", and "*forearm_twisting"* only is set for 200. For other joints such as "*lower_arm*" and "*upper_arm*" are set for 500 as shown in Image 24.

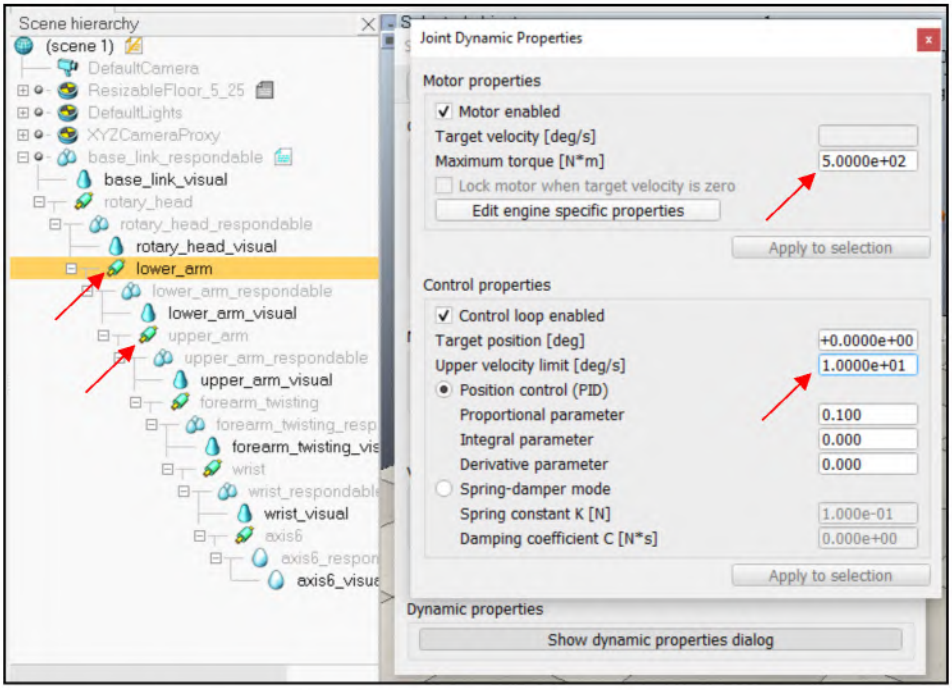


Image 24 Joint Settings for two joints only

B. Rigid Body ( )

Double click the icon ( ), click "*Show dynamic properties dialog*" and setting the rigid body as shown in Image 25 below. The setting is to make sure to check the box for "*Body is dynamic*". Numbers below the check box, do not change anything. Let the numbers is in default settings. Take a note the red arrows shown in the image below.

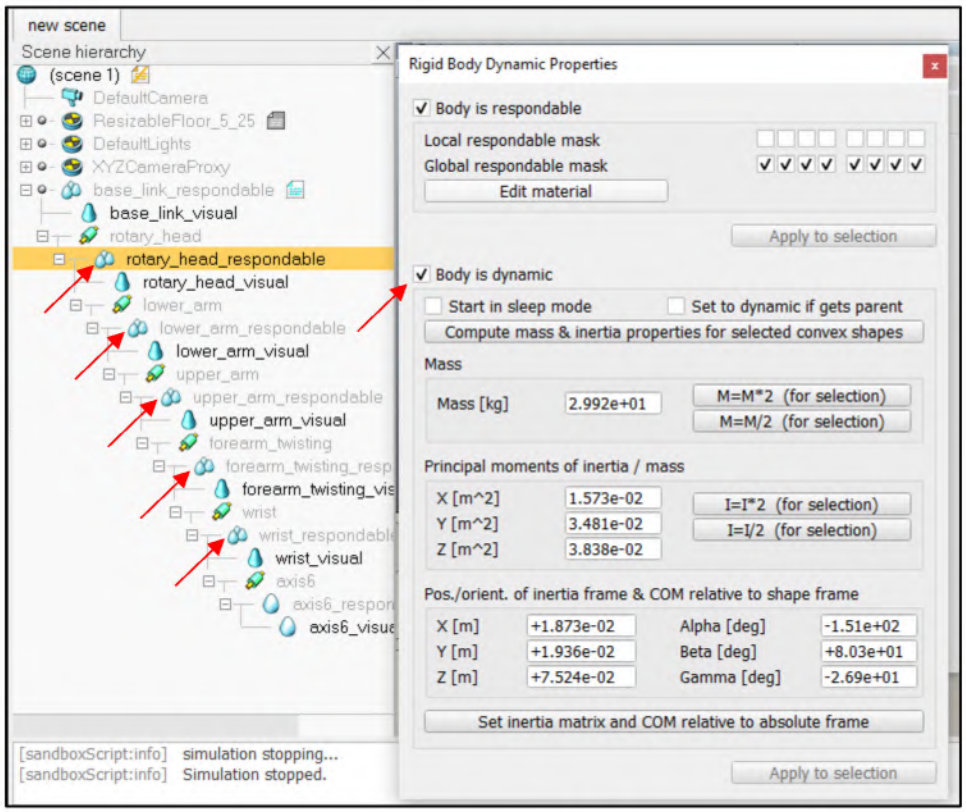


Image 25 Settings for Rigid Body

Finally, the settings are finished. The settings are a little bit complicated to understand them, because there are so many settings involve in the robotic arm model in CoppeliaSim software.

14. Important step! Double click the icon shown by the red arrow in Image 26. Delete all the programming code in the file.

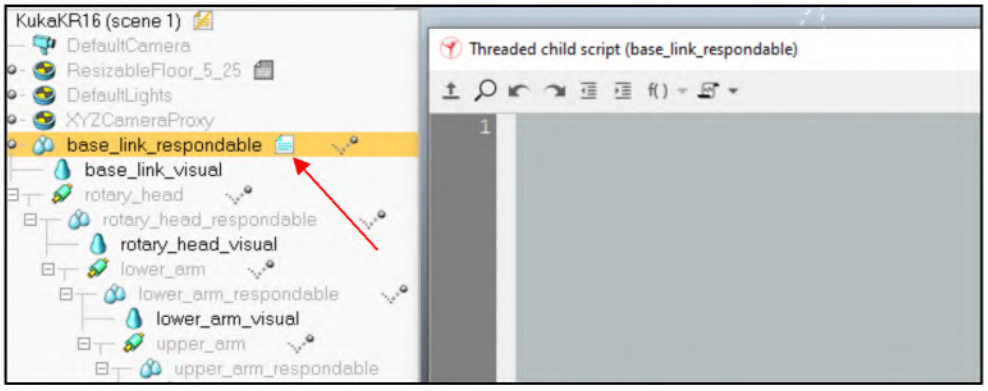


Image 26 Delete programming code

15. Next step, we can ignore the two file which are "*test.m*" and "*simpleTest.m*" because we will use new files for the next step of this tutorial. The new files already contained a Graphical User Interface created by me. Download the three remaining files from the Google Drive, which are "*DHmatrix.m*", "*fyp1.fig*", and "*fyp1.m*". I hope juniors will not claimed these files as their own works. Transfer these three files into the folder created in step 10. Refer image below.

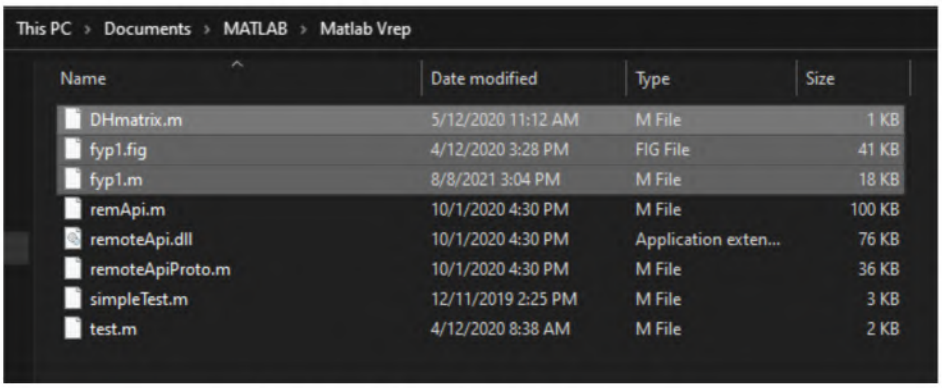


Image 27 Three New Files

16. Open "*fyp1.m*" in MATLAB software. Now, we will communicate between the file and CoppeliaSim just like we communicate in Step 12. Click "Start Simulation" in CoppeliaSim first, then click "Run Simulation" in the MATLAB. A GUI will appear. Try to enter any values in the empty boxes as shown in Image 28 and click "FK" button.

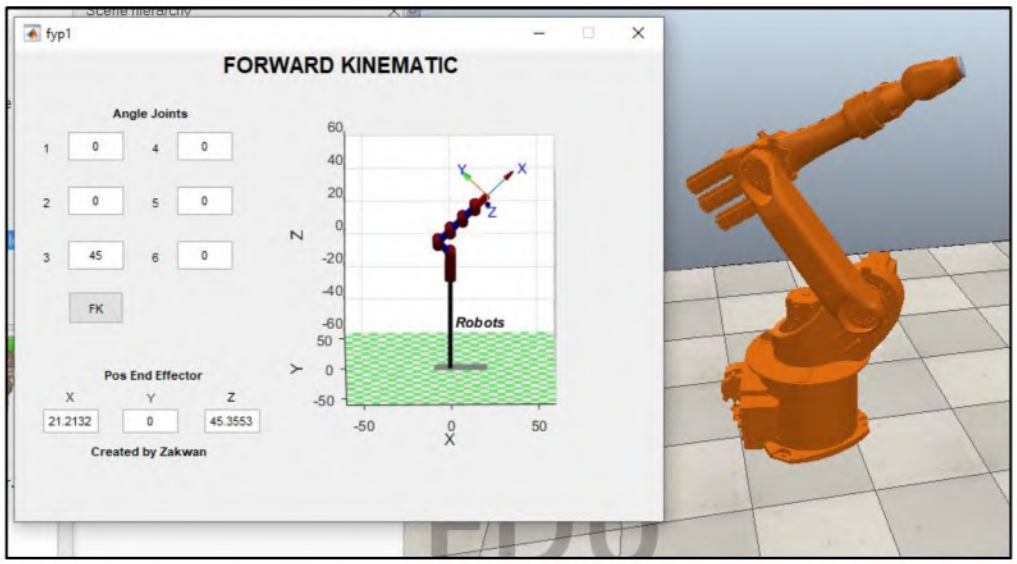


Image 28 Tutorial Finished

17. Finally, if you successfully get the results as shown in Image 28, you have successfully finished this tutorial. I hope you can understand this tutorial and I hope this tutorial can contribute a little bit for your projects.