

# Random movement

Follow the previous steps to install the virtual machine. After decompressing the system files, open the virtual machine and you can use it. The environment has been set up and the relevant code has been placed in the Ubuntu system, in the file path /home/dofbot/dofbot\_ws/src directory.

## 1. Start the robotic arm simulation

- **Virtual machine simulation starts**

Start MoveIT (virtual machine side)

```
roslaunch dofbot_config demo.launch
```

Open another terminal and enter the command line. (This program is simulated in rviz, the real machine will not move) (Virtual machine side)

```
roslaunch dofbot_moveit 01_set_move.py # python file
```

- **Real device startup**

```
roslaunch dofbot_config demo.launch #Virtual machine side
roslaunch dofbot_moveit 00_dofbot_move.py #Host side
roslaunch dofbot_moveit 01_set_move.py # python file #Virtual machine side
```

The corresponding relationship between the robotic arm servo and the joints: from the lowest end of the robotic arm to the end of the gripper.

Close case: [ctrl+c] to close. If it cannot be closed, execute [ctrl+z] again.

Key part of the program code description:

```
#Import header file
import rospy
from moveit_commander.move_group import MoveGroupCommander

#Initialize node
rospy.init_node("dofbot_random_move")
# Initialize the robot arm motion planning group
dofbot = MoveGroupCommander("dofbot") #Note: The group name here must be
consistent with the group name in the previous MoveIt configuration, otherwise it
will be invalid.
#Set random target point
dofbot.set_random_target()
# Set pose
dofbot.set_named_target("up")
dofbot.set_named_target("down") #Note: The up and down here are the postures
set during MoveIt configuration.

# Start exercising
dofbot.go()
sleep(0.5)
```

For detailed code, see `dofbot_ws/src/dofbot_moveit/scripts/01_set_move.py`

Experimental phenomenon: You can see that the robotic arm in rviz will randomly search for target points and move.

