

# Build-a-Bot 3.0

Senior PS | Round 1  
2-DoF Planar Plotter Simulation

## Introduction

My darlings, even a journalist of my stature cannot keep pace with the delicious scandals brewing at Hogwarts! I(Rita Skeeter) am commissioning you to design "The Virtual Scribe," a sophisticated two-degree-of-freedom apparatus to automate my most sensitive reporting. Whether you craft a rigid Cartesian frame or an elegant planar arm, your creation must translate commands into the fluid motions of a master calligrapher.

Within the enchanted realms of ROS2 and CoppeliaSim, you must prove your mechanical logic is as sharp as my Quick-Quotes Quill. Do be precise, pets—the wizarding world is waiting for its next exclusive, and I won't settle for anything less than perfection!

## Objective

Your task is to design and simulate a simple 2-DoF planar writing mechanism using ROS2 and CoppeliaSim/Gazebo.

The mechanism should mimic a basic plotter or handwriting robot capable of moving a pen tip along the X and Y axes. Control of both axes must be achieved via keyboard teleoperation, allowing real-time manual movement across the workspace.

## Core Tasks

You must successfully implement the following:

- Model Design: Create a 2-DoF mechanism in CoppeliaSim/Gazebo that enables motion in the X and Y directions. This can be achieved using:
  - Two orthogonal linear joints (Cartesian design), or
  - Two rotary joints in a planar arm configuration.
- ROS2 Integration: Set up a ROS2-CoppeliaSim/ROS2-Gazebo bridge that allows your ROS2 node to send control commands to the simulated robot's joints and (optionally) receive joint feedback.

Continued...

## Core Tasks

- **Teleoperation Control:** Implement a ROS2 node that subscribes to `geometry_msgs/Twist` messages from `teleop_twist_keyboard` (or an equivalent teleop package). Use:
  - `linear.x` for movement along the X-axis.
  - `linear.y` for movement along the Y-axis.
- **Control Logic:** Convert incoming Twist commands into appropriate position or velocity commands for your model's two actuated joints, ensuring responsive and stable motion
- **Demonstration:** You must be able to control the pen's position manually using the keyboard in real time. The simulation should display smooth bidirectional motion along both axes, enabling you to trace basic shapes or paths interactively.

## Technical Specifications

- Platform: ROS2 Humble
- Simulator: CoppeliaSim Edu, Gazebo Ignition
- Control Input: `teleop_twist_keyboard` (ROS2 built-in package)
- Key Topics (Suggested):
  - ``/cmd_vel`` (Type: `geometry_msgs/Twist`): For teleop input commands.

## Submission Details

- Submit a single ZIP file containing all relevant files.
- The ZIP must include:
  - Your ROS2 packages and .launch.py files.
  - The complete CoppeliaSim/Gazebo scene/model files (.ttt, .obj, etc.).
  - A well-documented README.md explaining setup, build, and execution steps.
- Include a link to a 2-minute unlisted YouTube video showing your simulation responding to teleop inputs in real time.

## Resources

For guidance and reference, participants may use the following official documentation:

- ROS2 Humble: <https://docs.ros.org/en/humble/index.html>
- CoppeliaSim Edu: <https://www.coppeliarobotics.com/>
- Gazebo: <https://gazebosim.org/docs/fortress/install/>

More Creative Solutions and Approaches are always welcomed!

All The Best!

## Judging Criteria

- Creativity
- Feasibility
- Implementation Quality
- Cost Effectiveness of Design

## For any Queries

Dev Kaushal

[outreach.parsec@iitdh.ac.in](mailto:outreach.parsec@iitdh.ac.in)