

BÁO CÁO GIỮA KỲ

Môn học: Lập trình Robot với ROS

1. Giới Thiệu

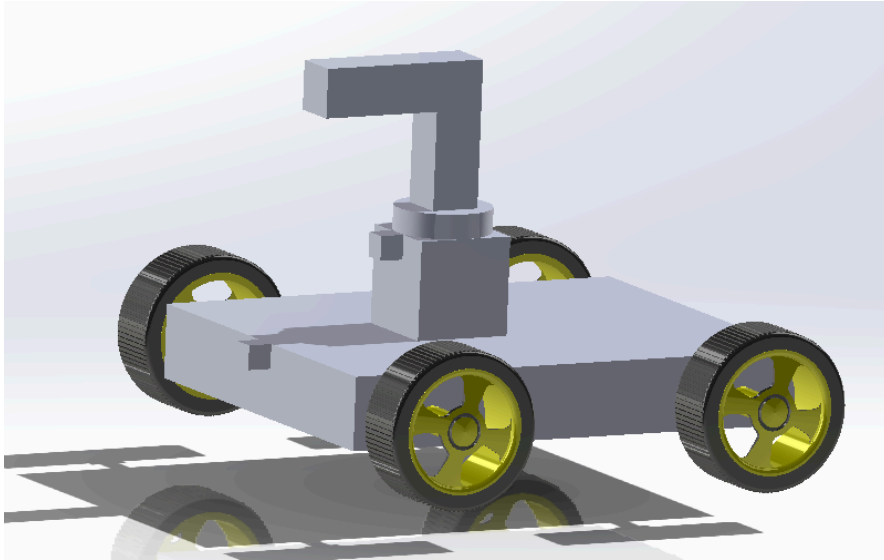
Dự án nhằm thiết kế mô hình xe car-like có tay máy tích hợp, mô phỏng trong ROS Gazebo, và điều khiển bằng ROS.

2. Dạng Robot, Động Học, Kích Thước

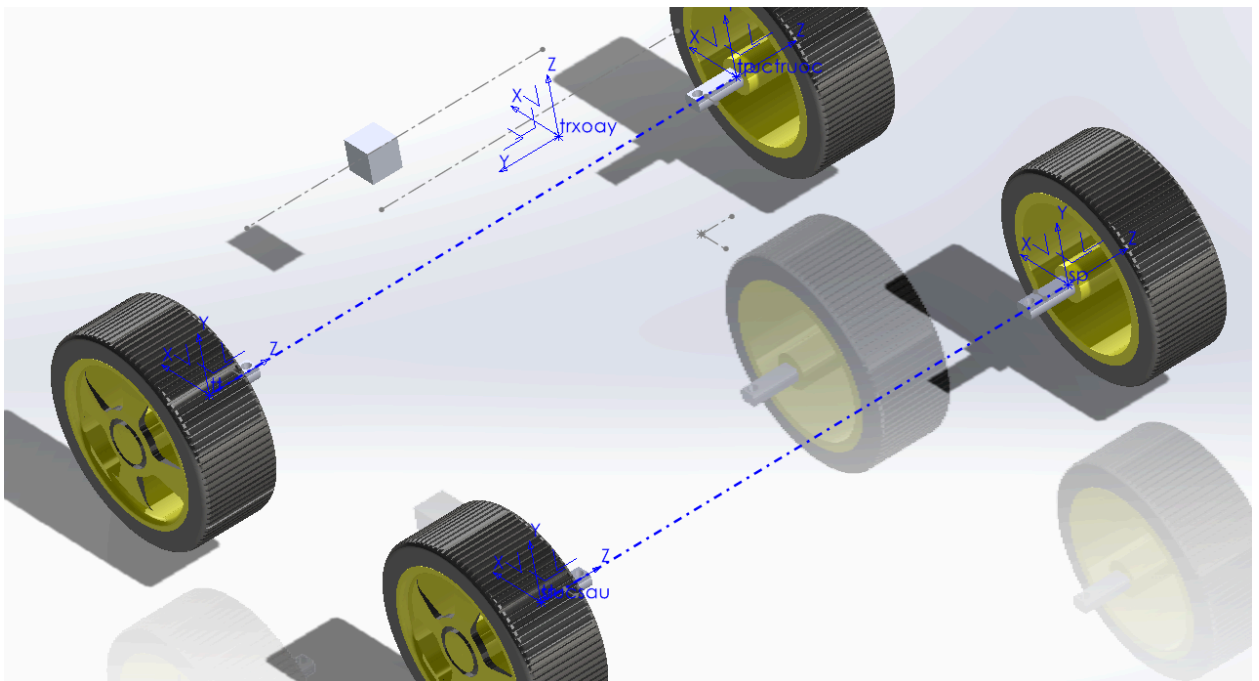
- **Dạng Robot:** Xe car-like có bốn bánh dẫn động.
- **Động Học:**
 - Chuyển động car-like theo cơ chế động học vi sai
 - Tay máy gồm hai bộ khớp: quay và tịnh tiến.
- **Thông số xe:**
 - Chiều dài: 0.2 m
 - Chiều rộng: 0.2 m
 - Chiều cao: 0.15 m
 - Khoảng cách giữa các trục trước và sau 0.1m
 - Tay máy: khớp quay 360 , tịnh tiến 0.05m
- **Cảm biến:**
 - **IMU:** Nhận thông tin gia tốc, quán tính.
 - **Camera:** Quan sát môi trường.
 - **Encoder:** Đo vận tốc bánh xe.

3. Thiết Kế SolidWorks, Hệ Trục Tọa Độ

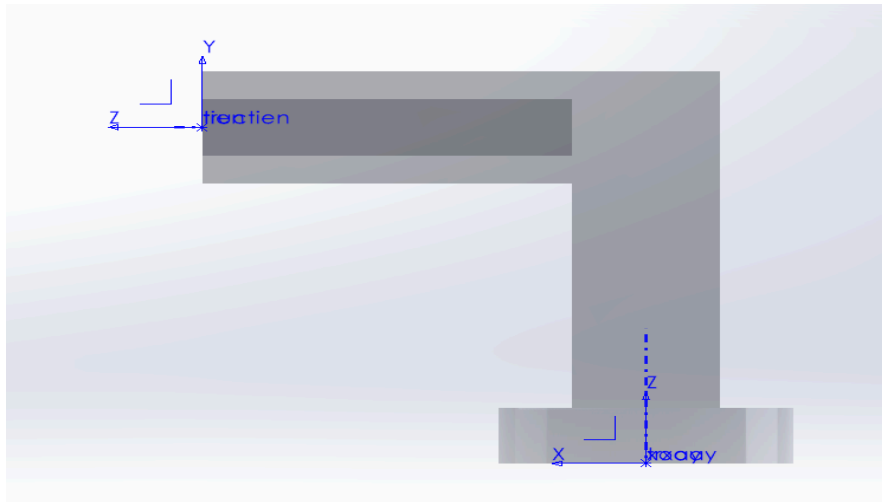
- Thiết kế hoàn chỉnh trong SolidWorks



- Cách đặt trục tọa độ
 - Tạo các trục trên phần chuyển động của bánh và tay máy. Trục ở giữa 2 cặp động cơ trước và sau, trục đi qua tâm của khớp xoay và trục theo chiều chuyển động của khớp tịnh tiến.
 - Gắn trục tọa độ vào bánh xe, chiều Z hướng ra bên ngoài theo chiều quay của bánh, X hướng về phía trước xe. Cả 4 bánh gắn trục đồng bộ nhau.



- Gắn trục tọa độ cho khớp nằm giữa vị trí gốc của khớp trên trục chuyển động. Z hướng theo chuyển động.



-
- Đặt trục tọa độ của cảm biến ở ngay vị trí cảm biến
- Đặt trục tọa độ của thân xe tại vị trí chính giữa chiều Z hướng lên trên, X hướng về phía trước đảm bảo xe không bị lật

4. Mô Tả File URDF

- **Liên kết (links):** Khung xe, bánh xe, tay máy.

- Base_link (Thân xe)
 - + Tt_link (Bánh trái trước)
 - + Tp_link (Bánh phải trước)
 - + Sp_link (Bánh phải sau)
 - + St_link (Bánh trái sau)
 - + Imu_link (IMU sensor)
 - + Camera_link (Camera)
 - + Xoay_link (Khớp xoay)
 - + Tien_link (Khớp tịnh tiến)

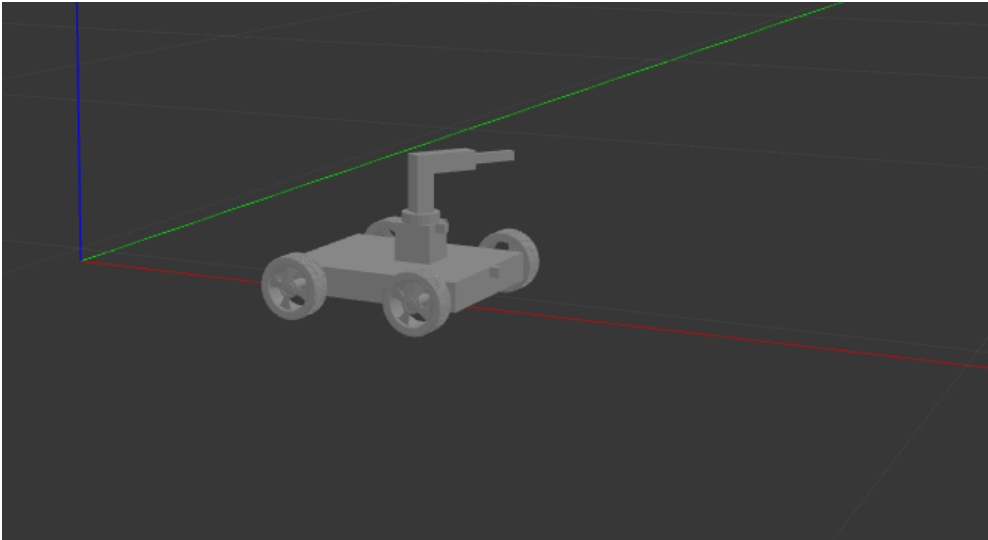
- **Liên kết (joints):**

- Joint type:
 - + Prismatic: khớp tịnh tiến
 - + Continuous: 4 bánh xe

- + Fix: thân xe và cảm biến
- + Revolute: khớp xoay:

5. Mô Tả Mô Phỏng Gazebo

- Mô hình URDF/SDF được load vào Gazebo.



- Điều khiển bánh xe.

```
kongfu@kongfu-Nitro-AN515-58: ~/catkin_ws

/home/kongfu/catkin_ws/src/cuoiky/lau... x kongfu@kongfu-Nitro-AN515-58: ~/catki... x
kongfu@kongfu-Nitro-AN515-58: ~/catkin_ws 38x2 kongfu@kongfu-Nitro-AN515-58: ~/catkin_ws 39x2
kongfu@kongfu-Nitro-AN515-58:~/catkin_ws$ rosrn cuoiky controller_node.py

Điêu khiển robot từ bàn phím:
-----
w : Tiến về phía trước
s : Lùi về phía sau
a : Quay trái
d : Quay phải
x : DỪNG robot
CTRL+C : Thoát
-----

[INFO] [1743564865.981564, 0.000000]:
Robot controller is running...

z: 0.0
---
linear:
  x: 0.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0
---
linear:
  x: 0.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0
---
```

- Điều khiển tay máy

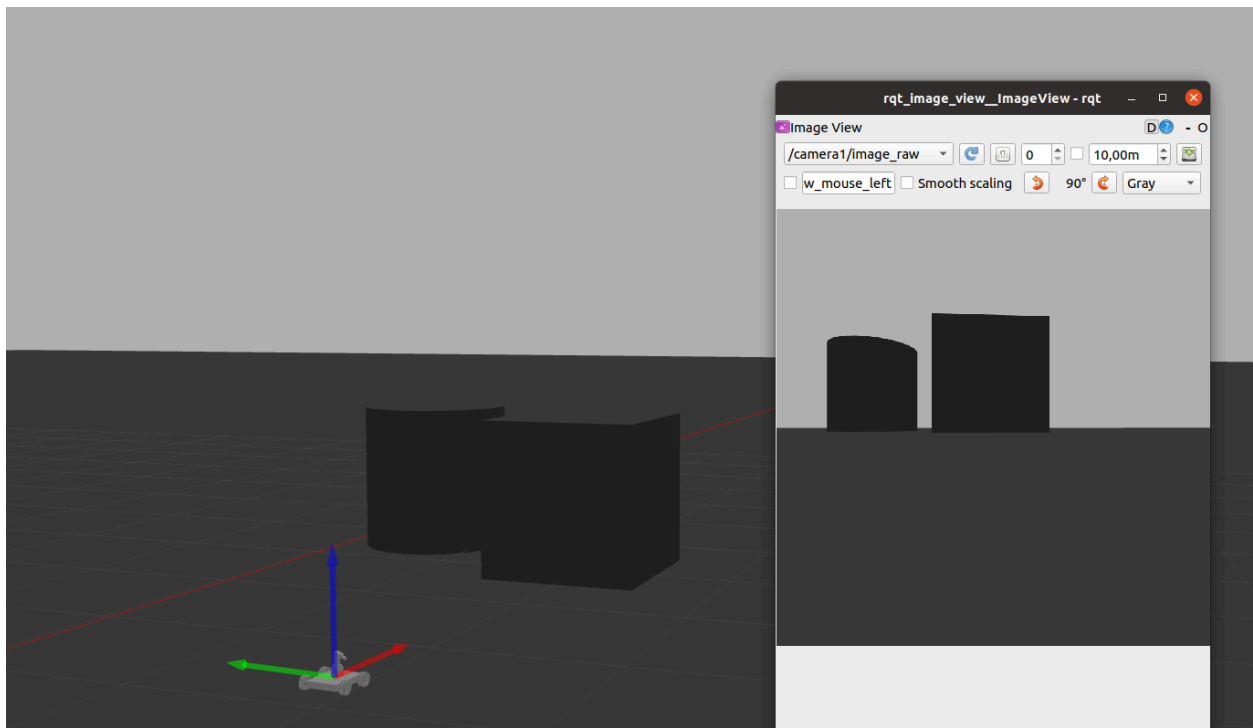
```
kongfu@kongfu-Nitro-AN515-58: ~/catkin_ws

roscore http://kongfu-... x /home/kongfu/catkin_... x kongfu@kongfu-Nitro-... x
kongfu@kongfu-Nitro-AN515-58: ~/catkin_ws 79x10
kongfu@kongfu-Nitro-AN515-58:~/catkin_ws$ rosrn cuoiky link_controller.py

Điêu khiển các link:
-----
w : Tiến về phía trước (tien_link)
s : Lùi về phía sau (tien_link)
a : Quay trái (xoay_link)
d : Quay phải (xoay_link)
x : DỪNG
CTRL+C : Thoát

kongfu@kongfu-Nitro-AN515-58: ~/catkin_ws 38x10 kongfu@kongfu-Nitro-AN515-58: ~/catkin_ws 39x10
```

- Đọc dữ liệu từ cảm biến.
- - Camera



- IMU

6. Cơ Chế Điều Khiển

- **ROS Nodes:**
 - Node nhận tín hiệu từ ROS Topic (/cmd_vel) để điều khiển.
 - Node xử lý dữ liệu cảm biến, điều chỉnh tay máy.

7. Các Thành Phần Chính Của Code, Structure Folder

- **urdf/**: Chứa mô hình robot.
- **scripts/**: Code điều khiển và xử lý dữ liệu.
- **meshes /**: chứa các
- **launch/**: File launch để khởi động simulation.
- **config/**: Tham số PID, cấu hình.
- Plugin cảm biến
 - + IMU

```
<gazebo reference="imu_link">
  <gravity>true</gravity>
  <sensor name="imu_sensor" type="imu">
    <always_on>true</always_on>
    <update_rate>100</update_rate>
    <visualize>true</visualize>
    <topic>imu/data</topic>
    <plugin filename="libgazebo_ros_imu_sensor.so" name="imu_plugin">
      <topicName>imu</topicName>
      <bodyName>imu_link</bodyName>
      <updateRateHZ>10.0</updateRateHZ>
      <gaussianNoise>0.0</gaussianNoise>
      <xyzOffset>0 0 0</xyzOffset>
      <rpyOffset>0 0 0</rpyOffset>
      <frameName>imu_link</frameName>
      <initialOrientationAsReference>false</initialOrientationAsReference>
    </plugin>
    <pose>0 0 0 0 0 0</pose>
  </sensor>
</gazebo>
```

+ Camera


```

<!-- Cảm biến camera -->
<gazebo reference="camera_link">
  <sensor type="camera" name="camera1">
    <update_rate>30.0</update_rate>
    <camera name="head">
      <horizontal_fov>1.3962634</horizontal_fov>
      <image>
        <width>800</width>
        <height>800</height>
        <format>R8G8B8</format>
      </image>
      <clip>
        <near>0.02</near>
        <far>300</far>
      </clip>
      <noise>
        <type>gaussian</type>
        <mean>0.0</mean>
        <stddev>0.007</stddev>
      </noise>
    </camera>
    <plugin name="camera_controller" filename="libgazebo_ros_camera.so">
      <alwaysOn>true</alwaysOn>
      <updateRate>0.0</updateRate>
      <cameraName>xe/camera1</cameraName>
      <imageTopicName>image_raw</imageTopicName>
      <cameraInfoTopicName>camera_info</cameraInfoTopicName>
      <frameName>camera_link</frameName>
      <hackBaseline>0.07</hackBaseline>
      <distortionK1>0.0</distortionK1>
      <distortionK2>0.0</distortionK2>
      <distortionK3>0.0</distortionK3>
      <distortionT1>0.0</distortionT1>
      <distortionT2>0.0</distortionT2>
    </plugin>
  </sensor>
</gazebo>

```

Điều khiển xe (Robot 4 bánh):

- Sử dụng các bánh xe điều khiển độc lập để di chuyển và quay.
Các lệnh điều khiển:

- w: Tiến
- s: Lùi
- a: Quay trái
- d: Quay phải
- x: Dừng
- Mã điều khiển sử dụng ROS Publisher để gửi các giá trị vận tốc cho các bánh xe.

Điều khiển tay máy (Khớp tay và cánh tay):

- Khớp xoay (Revolute) và Khớp tịnh tiến (Prismatic) điều khiển các chuyển động của tay máy.
- Các lệnh điều khiển tay máy:
 - w: Tiến
 - s: Lùi
 - a: Quay trái
 - d: Quay phải
- Mã điều khiển gửi các lệnh tới các joint controllers để điều chỉnh chuyển động của tay máy.

8. Kết Luận

Báo cáo trình bày thiết kế xe car-like và quá trình tổ chức mô phỏng trong ROS Gazebo. Chưa thể điều khiển xe và tay máy chính xác như ý muốn và các dữ liệu cảm biến còn thiếu