Robot Subsystems

7th January 2020 at 9:02am

Actuation

- Mobile Platform
- Manipulator

Sensing

- Encoders
- Laser Scanners
- Cameras
- Depth Cameras

Computing

- Arduino Boards
- SBC - RPi, Beaglebone
- Mini-CPUs - Intel NUC

Mobile Dojo - Levels of motion control

7th January 2020 at 9:11am

Level 0 - Motors, Wheels and Encoders

Level 1 - Motor Controllers and Drivers

Level 2 - ROS Base Controller

Level 3 - Frame-Base Motion using the move_base ROS Package

Level 4 - SLAM using the gmapping and amcl ROS Packages

Level 5 - Semantic Goals

Twisting and Turning with ROS - Mobile Robot Dynamics

7th January 2020 at 10:07am

Understanding Twist message

`rosmsg show geometry_msgs/Twist`

```
geometry_msgs/Vector3 linear
float64 x
float64 y
float64 z
geometry_msgs/Vector3 angular
float64 x
float64 y
float64 z
```

Example Twist Message

> Move straight ahead with a speed of 0.1 meters per second

`'{linear: {x: 0.1, y: 0, z: 0}, angular: {x: 0, y: 0, z: 0}}'`

> Rotate counterclockwise with an angular velocity of 1.0 radians per second

`{linear: {x: 0, y: 0, z: 0}, angular: {x: 0, y: 0, z: 1.0}}`

Challenge: Merry go Round

7th January 2020 at 3:48pm

> Move the turtlebot to go in a circle of radius 0.5 meter with the linear velocity of 0.2 meter per second

To do:

- Bring turtlebot3 and rviz tool by launching following command

`roslaunch turtlebot3_fake turtlebot3_fake.launch`

- Publish the velocity command with appropriate value for `/cmd_vel` topic

```
rostopic pub -r 10 /cmd_vel geometry_msgs/Twist '{linear: {x: 0.2, y: 0, z: 0}, angular: {x: 0, y: 0, z: ?}}'
```

Touch and Back : Running robot through python Script

7th January 2020 at 11:01am

Creating `Twist()` messages in python

```
move_cmd = Twist()

move_cmd.linear.x = 0.2
```

```
move_cmd.angular.z = 1.0
```