TURTLEBOT3

Burger

1부

PC Setup

PC Setup/Ubuntu

- ROS1

  Raspberry Pi 3B+
  Jetson Nano does not support ROS Kinectic

  Kinetic : https://releases.ubuntu.com/16.04.7/

- ROS2

  Raspberry Pi 4 4G

  Humble : https://releases.ubuntu.com/22.04/

PC Setup/ROS Install

- ROS1

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ wget https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_kinetic.sh
$ 755 ./install_ros_kinetic.sh
$ bash ./install_ros_kinetic.sh
```

- ROS2

```

```

PC Setup/ROS Dependent Package

- ROS1

```
$ sudo apt-get install ros-kinetic-joy ros-kinetic-teleop-twist-joy \
ros-kinetic-teleop-twist-keyboard ros-kinetic-laser-proc \
ros-kinetic-rgbd-launch ros-kinetic-depthimage-to-laserscan \
ros-kinetic-rosserial-arduino ros-kinetic-rosserial-python \
ros-kinetic-rosserial-server ros-kinetic-rosserial-client \
ros-kinetic-rosserial-msgs ros-kinetic-amcl ros-kinetic-map-server \
ros-kinetic-move-base ros-kinetic-urdf ros-kinetic-xacro \
ros-kinetic-compressed-image-transport ros-kinetic-rqt* \
ros-kinetic-gmapping ros-kinetic-navigation ros-kinetic-interactive-markers
```

- ROS2

```
$ sudo apt ros-humble-gazebo-*
```

```
$ sudo apt ros-humble-cartographer
$ sudo apt ros-humble-cartographer-ros
```

```
$ sudo apt ros-humble-navigation2
$ sudo apt ros-humble-nav2-bringup
```

PC Setup/Turtlebot Package

- ROS1

```
$ sudo apt-get ros-kinetic-dynamixel-sdk
$ sudo apt-get ros-kinetic-turtlebot3-msgs
$ sudo apt-get ros-kinetic-turtlebot3
```

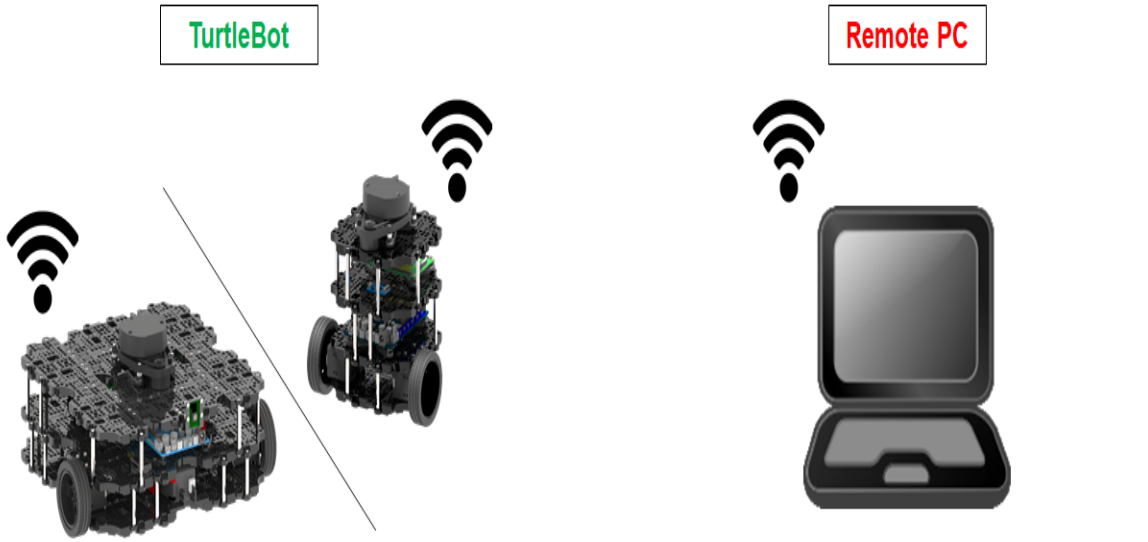
- ROS2

```
$ source ~/.bashrc
$ sudo apt ros-humble-dynamixel-sdk
$ sudo apt ros-humble-turtlebot3-msgs
$ sudo apt ros-humble-turtlebot3
```

PC Setup/**Environment Configuration**

- ROS1

```
$ echo "export TURTLEBOT3_MODEL=burger" >> ~/.bashrc
```



ROS_MASTER_URI = http://IP_OF_REMOTE_PC:11311
ROS_HOSTNAME = IP_OF_TURTLEBOT

ROS_MASTER_URI = http://IP_OF_REMOTE_PC:11311
ROS_HOSTNAME = IP_OF_REMOTE_PC

* Example when ROS Master is running on the Remote PC

```
if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

if [ -x /usr/bin/mint-fortune ]; then
    /usr/bin/mint-fortune
fi

alias eb='nano ~/.bashrc'
alias sb='source ~/.bashrc'
alias gs='git status'
alias gp='git pull'
alias cw='cd ~/catkin_ws'
alias cs='cd ~/catkin_ws/src'
alias cm='cd ~/catkin_ws && catkin_make'

source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash

export ROS_MASTER_URI=http://192.168.0.100:11311
export ROS_HOSTNAME=192.168.0.100

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell ^_ Go To Line
```

PC Setup/**Environment Configuration**

- ROS2

```
$ echo 'export ROS_DOMAIN_ID=30 #TURTLEBOT3' >> ~/.bashrc
$ source ~/.bashrc
```