

# Table of Contents

<b>Preface</b>	<b>xi</b>
<b>Chapter 1: Getting Started with ROS</b>	<b>1</b>
<b>What does ROS do and what are the benefits of learning ROS?</b>	<b>2</b>
Who controls ROS?	3
<b>Which robots are using ROS?</b>	<b>3</b>
<b>Installing and launching ROS</b>	<b>5</b>
Configuring your Ubuntu repositories	6
Setting up your sources.list file	6
Setting up your keys	7
Installing ROS Kinetic	7
Initialize rosdep	7
Environment setup	8
Getting rosinstall	8
Troubleshooting – examining your ROS environment	8
<b>Creating a catkin workspace</b>	<b>9</b>
<b>ROS packages and manifest</b>	<b>10</b>
ROS manifest	11
Exploring the ROS packages	11
rospack find packages	12
rospack list	12
<b>ROS nodes, topics, and messages</b>	<b>13</b>
ROS nodes	14
ROS topics	14
ROS messages	15
ROS Master	15
Invoking the ROS Master using roscore	16
ROS commands to determine the nodes and topics	17

<b>Turtlesim – the first ROS robot simulation</b>	<b>19</b>
Starting turtlesim nodes	19
roslaunch command	20
Turtlesim nodes	21
Turtlesim topics and messages	23
rostopic list	24
rostopic type	24
rosmmsg list	24
rosmmsg show	25
rostopic echo	25
Move the turtle by publishing /turtle1/cmd_vel	26
Move the turtle using the keyboard or joystick	28
Parameter Server of Turtlesim	28
rosparam help	28
rosparam list for the /turtlesim node	29
Change parameters for the color of the turtle's background	29
ROS services to move turtle	30
rosservice call	31
<b>ROS commands summary</b>	<b>32</b>
<b>Summary</b>	<b>33</b>
<b>Chapter 2: Creating Your First Two-Wheeled ROS Robot (in Simulation)</b>	<b>35</b>
<b>Introducing rviz</b>	<b>36</b>
Installing and launching rviz	36
Getting familiar with rviz	38
Displays panel	38
Views and Time panels	39
Toolbar	40
Main window menu bar	41
<b>Creating and building a ROS package</b>	<b>41</b>
<b>Building a differential drive robot URDF</b>	<b>42</b>
Creating a robot chassis	43
Using roslaunch	44
Adding wheels	48
Adding a caster	51
Adding color	52
Adding collisions	54
Moving the wheels	55
A word about tf and robot_state_publisher	57
Adding physical properties	57
Trying URDF tools	59
check_urdf	59
urdf_to_graphviz	59

<b>Gazebo</b>	<b>60</b>
Installing and launching Gazebo	61
Using roslaunch with Gazebo	62
Getting familiar with Gazebo	63
Environment toolbar	64
World, Insert and Layers panels	66
Joints panel	67
Main window menu bar	68
Simulation panel	68
Modifications to the robot URDF	68
Adding the Gazebo tag	69
Specifying color in Gazebo	69
A word about the <visual> and <collision> elements in Gazebo	69
Verifying a Gazebo model	70
Viewing the URDF in Gazebo	70
Tweaking your model	72
Moving your model around	73
Other ROS simulation environments	74
<b>Summary</b>	<b>75</b>
<b>Chapter 3: Driving Around with TurtleBot</b>	<b>77</b>
<b>Introducing TurtleBot 2</b>	<b>78</b>
<b>Loading TurtleBot 2 simulator software</b>	<b>80</b>
<b>Launching TurtleBot 2 simulator in Gazebo</b>	<b>81</b>
Problems and troubleshooting	83
ROS commands and Gazebo	84
Keyboard teleoperation of TurtleBot 2 in simulation	86
<b>Setting up to control a real TurtleBot 2</b>	<b>87</b>
TurtleBot 2 standalone test	88
<b>Networking the netbook and remote computer</b>	<b>90</b>
Types of networks	90
Network addresses	91
Remote computer network setup	92
Netbook network setup	92
Secure Shell (SSH) connection	93
Summary of network setup	93
Troubleshooting your network connection	94
Testing the TurtleBot 2 system	94
<b>TurtleBot 2 hardware specifications</b>	<b>96</b>
TurtleBot 2 dashboard	96
<b>Moving the real TurtleBot 2</b>	<b>98</b>
Using keyboard teleoperation to move TurtleBot 2	98
Using ROS commands to move TurtleBot 2 around	100

Writing your first Python script to control TurtleBot 2	101
<b>Introducing rqt tools</b>	<b>104</b>
rqt_graph	105
rqt message publisher and topic monitor	107
<b>TurtleBot's odometry</b>	<b>110</b>
Odom for the simulated TurtleBot 2	114
Real TurtleBot 2's odometry display in rviz	117
<b>TurtleBot 2 automatic docking</b>	<b>118</b>
<b>Introducing TurtleBot 3</b>	<b>120</b>
<b>Loading TurtleBot 3 simulation software</b>	<b>122</b>
<b>Launching TurtleBot 3 simulation in rviz</b>	<b>123</b>
<b>Launching TurtleBot 3 simulation in Gazebo</b>	<b>125</b>
<b>Hardware assembly and testing</b>	<b>129</b>
<b>Loading TurtleBot 3 software</b>	<b>129</b>
Installing remote computer software	129
Installing SBC software	130
Loading Ubuntu MATE	131
Loading ROS packages	133
Loading TurtleBot 3 packages	134
Setting up udev rules for TurtleBot 3	135
<b>Networking TurtleBot 3 and the remote computer</b>	<b>135</b>
Remote computer network setup	136
TurtleBot 3 network setup	137
SSH connection	138
Testing the SSH communication	139
Troubleshooting your network connection	139
<b>Moving the real TurtleBot 3</b>	<b>140</b>
Using keyboard teleoperation to move TurtleBot 3	141
<b>Summary</b>	<b>143</b>
<b>Chapter 4: Navigating the World with TurtleBot</b>	<b>145</b>
<b>3D vision systems for TurtleBot</b>	<b>146</b>
How these 3D vision sensors work	146
Comparison of 3D sensors	147
Microsoft Kinect	147
ASUS	149
PrimeSense Carmine	152
Intel RealSense	153
Hitachi-LG LDS	157
Obstacle avoidance drawbacks	158
<b>Configuring TurtleBot and installing the 3D sensor software</b>	<b>158</b>
Kinect	159
ASUS and PrimeSense	159

Intel RealSense	160
Camera software structure	161
Defining terms	161
<b>Testing the 3D sensor in standalone mode</b>	<b>162</b>
<b>Running ROS nodes for visualization</b>	<b>163</b>
Visual data using Image Viewer	163
Visual data using rviz	165
<b>Navigating with TurtleBot</b>	<b>169</b>
Mapping a room with TurtleBot 2	170
Defining terms	170
Building a map	170
How does TurtleBot accomplish this mapping task?	174
Autonomous navigation with TurtleBot 2	175
Defining terms	176
Driving without steering TurtleBot 2	176
rviz control	178
How does TurtleBot accomplish this navigation task?	181
Navigating to a designated location	183
Navigating to waypoints with a Python script using a map	185
Defining TurtleBot's position on a map	187
Defining waypoints on a map	188
Using Python code to move TurtleBot	189
TurtleBot at final goal point	191
SLAM for TurtleBot 3	193
Autonomous navigation with TurtleBot 3	195
rqt_reconfigure	197
Exploring ROS navigation further	199
<b>Summary</b>	<b>200</b>
<b>Chapter 5: Creating Your First Robot Arm (in Simulation)</b>	<b>201</b>
<b>Features of Xacro</b>	<b>202</b>
<b>Building an articulated robot arm URDF using Xacro</b>	<b>203</b>
Specifying a namespace	203
Using the Xacro property tag	204
Expanding Xacro	207
Using roslaunch for rrbot	208
Using the Xacro include and macro tags	211
Adding mesh to the robot arm	214
<b>Controlling an articulated robot arm in Gazebo</b>	<b>220</b>
Adding Gazebo-specific elements	221
Fixing the robot arm to the world	222
Viewing the robot arm in Gazebo	223
Adding controls to Xacro	225
Defining transmission elements for joints	225

Adding a Gazebo ROS control plugin	226
Creating a YAML configuration file	227
Creating a control launch file	228
Controlling your robot arm with the ROS command line	229
Controlling your robot arm with rqt	232
Trying more things in rqt	234
<b>Summary</b>	<b>235</b>
<b>Chapter 6: Wobbling Robot Arms Using Joint Control</b>	<b>237</b>
<b>Introducing Baxter</b>	<b>238</b>
Baxter, the research robot	241
Baxter Simulator	242
<b>Baxter's arms</b>	<b>243</b>
Baxter's bend joints	244
Baxter's twist joints	245
Baxter's coordinate frame	246
Control modes for Baxter's arms	247
Baxter's grippers	248
Baxter's arm sensors	249
<b>Loading Baxter software</b>	<b>250</b>
Installing Baxter SDK software	250
Installing Baxter Simulator	252
Configuring the Baxter shell	253
Installing MoveIt!	254
<b>Launching Baxter Simulator in Gazebo</b>	<b>255</b>
Bringing Baxter Simulator to life	256
Warm-up exercises	261
Flexing Baxter's arms	263
Untucking Baxter's arms	263
Wobbling arms	265
Controlling arms and grippers with a keyboard	266
Controlling arms and grippers with a joystick	267
Controlling arms with a Python script	270
Recording and replaying arm movements	272
<b>Baxter's arms and forward kinematics</b>	<b>273</b>
Joints and joint state publisher	273
Understanding tf	278
A program to move Baxter's arms to a zero angle position	279
Commanding the joint angles directly	281
rviz tf frames	282
Viewing a tf tree of robot elements	283
<b>Introducing MoveIt!</b>	<b>284</b>
Planning a move of Baxter's arms with MoveIt!	286

Adding objects to a scene	288
Position of objects	290
Planning a move to avoid obstacles with MoveIt!	290
<b>Configuring a real Baxter setup</b>	<b>292</b>
<b>Controlling a real Baxter</b>	<b>294</b>
Commanding joint position waypoints	295
Commanding joint torque springs	295
Demonstrating joint velocity	297
Additional examples	297
Visual servoing and grasping	297
<b>Inverse kinematics</b>	<b>298</b>
Moving Baxter's arms with IK	299
<b>Using a state machine to perform YMCA</b>	<b>303</b>
<b>Summary</b>	<b>308</b>
<b>Chapter 7: Making a Robot Fly</b>	<b>311</b>
<b>Introducing quadrotors</b>	<b>312</b>
Why are quadrotors so popular?	313
Defining roll, pitch, and yaw	313
How do quadrotors fly?	314
Components of a quadrotor	316
Adding sensors	317
Quadrotor communications	317
<b>Understanding quadrotor sensors</b>	<b>318</b>
Inertial measurement unit	318
Quadrotor condition sensors	318
<b>Preparing to fly your quadrotor</b>	<b>319</b>
Testing your quadrotor	319
Pre-flight checklist	320
Precautions when flying your quadrotor	320
Following the rules and regulations	321
<b>Using ROS with UAVs</b>	<b>321</b>
<b>Introducing Hector Quadrotor</b>	<b>322</b>
Loading Hector Quadrotor	323
Launching Hector Quadrotor in Gazebo	325
Flying Hector outdoors	325
Flying Hector indoors	330
<b>Introducing Crazyflie 2.0</b>	<b>334</b>
Controlling Crazyflie without ROS	336
Communicating using Crazyradio PA	337
Loading Crazyflie ROS software	338
Setting up udev rules for Crazyradio	340

Pre-flight check	341
Flying Crazyflie with teleop	342
Details of teleop_xbox360.launch	344
Flying with a motion capture system	346
Flying multiple Crazyflies	346
<b>Introducing Bebop</b>	<b>347</b>
Loading bebop_autonomy software	349
Testing Bebop communications	351
Flying Bebop using commands	352
Take off	352
Landing	353
<b>Summary</b>	<b>353</b>
<b>Chapter 8: Controlling Your Robots with External Devices</b>	<b>355</b>
<b>Creating a custom ROS game controller interface</b>	<b>356</b>
Testing a game controller	357
Alternative test of a game controller	358
Using the ROS joy package	360
Controlling Turtlesim with a custom game controller interface	360
<b>Creating a custom ROS Android device interface</b>	<b>366</b>
Installing Android Studio and tools	367
Installing a ROS–Android development environment	368
Defining terms	370
Introducing ROS–Android development	370
<b>Creating ROS nodes on Arduino or Raspberry Pi</b>	<b>371</b>
Using Arduino	372
Installing Arduino IDE software	372
Installing ROS–Arduino software	373
Ultrasonic sensor control using ROS and Arduino	378
Other applications using ROS and Arduino	386
Using Raspberry Pi	387
Installing ROS on the Raspberry Pi	387
<b>Summary</b>	<b>388</b>
<b>Chapter 9: Flying a Mission with Crazyflie</b>	<b>389</b>
<b>Mission components</b>	<b>391</b>
Kinect for Windows v2	391
Crazyflie operation	392
Mission software structure	392
OpenCV and ROS	394
<b>Loading software for the mission</b>	<b>395</b>
Installing libfreenect2	396
Installing iai_kinect2	399

Using the iai_kinect2 metapackage	402
kinect2_bridge and kinect2_viewer	402
kinect2_calibration	403
<b>Setting up the mission</b>	<b>408</b>
Detecting Crazyflie and a target	409
Identifying markers in a color image	409
Detecting and viewing markers with OpenCV	412
Using Kinect and OpenCV	414
How to track Crazyflie	417
<b>How to control Crazyflie</b>	<b>420</b>
Crazyflie control states	420
Using ROS services to control takeoff and land	421
Using PID control for hover and flight	423
Using an observer node	425
<b>Flying Crazyflie</b>	<b>426</b>
Hovering in place	426
What makes hover work?	427
Flying to a stationary target	428
What makes target detection work?	430
Learned lessons	431
Logging messages with rosout and rospy	431
<b>Summary</b>	<b>432</b>
<b>Chapter 10: Controlling Baxter with MATLAB®</b>	<b>433</b>
<b>Installing the MATLAB Robotics System Toolbox</b>	<b>434</b>
Check the MATLAB and Robotics System Toolbox versions	434
View the Robotics System Toolbox commands for ROS	434
<b>Using MATLAB Robotics System Toolbox and Baxter Simulator</b>	<b>435</b>
Installing Baxter messages in MATLAB	435
Running Baxter Simulator and MATLAB	437
Making Baxter move	440
<b>Summary</b>	<b>444</b>
<b>Index</b>	<b>445</b>