PR2 User Manual

Willow Garage

October 5, 2012

Contents

CHAPTER 1

Manual Overview

This manual is intended to give PR2 users information to successfully install, use, and develop code on the PR2 robot. The software on the PR2 is based on ROS. Please visit ros.org, to learn about ROS and higher-level software available for the PR2. To get started with the PR2 as quickly as possible; please start by reading Chapter 2: Safety, do not start operating the PR2 before reading Chapter 2. If the PR2 is being set up for the first time, please see the tutorial video "Unpacking the PR2" to learn how to set it up properly before it is used, then feel free to move on to Chapter 6 to learn how to start-up and run the PR2.

1.1 Before you start

Before getting started, please read through this section to review everything needed to use the PR2 safely and successfully.

- **PR2 Safety**
  - Be sure to read Chapter 2: Safety in its entirety before using the PR2.
- **Safe Environment**
  - The space where the PR2 operates should be free of hazards. Specifically, stairways and other drop offs can pose an extreme danger. Avoid hazardous objects, sharp objects (such as knives), sources of fire, hazardous chemicals, or furniture that could be knocked over. See Chapter 2: Safety, for more details on making sure the PR2's users and environment is safe.
- **Space**
  - To use the PR2, enough space is needed for it to drive around, move its arms, and perform tasks. The PR2 is designed to move through ADA-compliant spaces (Americans with Disabilities Act). Doorways should be at least 32 inches wide and the ground should be flat and level.

- **Development Tools**
  - At least one laptop or desktop computer to connect to the robot is needed. The PR2 ships with a base-station computer, which is a desktop, but PR2 users must provide their own screen, mouse, and keyboard. A laptop with wireless access is ideal.
- **Linux**
  - Familiarity with the Linux command-line is strongly recommended. The computers in the base of the PR2, as well as the base-station computer, run Ubuntu Linux. Since the computers in the robot do not have attached displays, all tasks on them have to be performed by logging in remotely (e.g. via ssh).
- **Electrical**
  - The PR2 charges using a standard 120V or 220 VAC power outlet. The robot can draw 15A of current when plugged in, so it is strongly recommend to recharge the PR2 only on outlets with no other devices on the same circuit breaker.
- **ROS**
  - All of the PR2 software is based on ROS. Completing the beginner tutorials available at ros.org will help PR2 users understand the structure of the software on the robot and will give PR2 users the tools to understand the flow of data in the system.
- **PR2 Support**
  - Please visit the PR2 support web page at pr2support.willowgarage.com to review beginner tutorials, service information, modularity specification, safety information, and to submit a ticket or report a safety incident.

CHAPTER 2

Safety

Safety is a key goal at Willow Garage. Safety is important yet challenging, and it is a continual process shared by the robot designer, user, and administrator. The following chapter provides an overview of the issues, describes some safety-related design features, enumerates a set of basic usage guidelines to support safety, and details the safety program.

2.1 Safety Overview

Safety is a vitally important concern whenever anyone is around a PR2. It is a heavy piece of equipment with many moving parts. The robot travels through the environment and can carry and manipulate a wide variety of objects. Its movements and actions are not completely predictable. The PR2 can cause significant damage if it falls on or runs over someone. There are several ways it can pinch, grab, and twist fingers or other body parts. The robot can wield dangerous implements and knock heavy objects over. People must always be cautious and attentive when around a PR2.

Managing safety is a challenge when dealing with any complex engineering system. In the case of the PR2, consider also the open, extensible, programmable and experimental nature of the platform. The PR2's capabilities and behaviors change over time, with user interactions, and with reprogramming. With this in mind, Willow Garage has chosen a three-fold approach. First, the PR2 is designed to minimize potential risks and maximize inherent safety. Second, all users receive communication about how to minimize risk. And third, an explicit safety program ensures that the community continues to identify potential hazards, seek design mitigations, and communicate effective usage guidelines.

2.2 Design Features

While retaining the power of an open platform, both hardware and software are designed to minimize risks. These aspects of the design are often described as inherent safety features. For example, the PR2's arms are back-drivable. This means that when an arm encounters an object, be it a table or a person, the interaction will drive the motors back and bring the arm to a stop. Unlike traditional industrial robots, the PR2's arms cannot "punch through" an object.

The PR2's arms have been designed using relatively small motors with respect to their payload. This is possible due to a spring counterbalance offsetting the gravity forces acting on the arms. That is, the arms do not need to hold their own weight against gravity, so the motors need only be strong enough to hold the payload. The arms simply cannot push very hard.

In software, low-level checking has been incorporated to limit the current in a motor, the velocity of a motor, and the range through which a joint should travel. Users and developers are obviously discouraged from changing these configurations. High-level applications also avoid obstacles during navigation and movement, using the various on-board sensors.

These design choices help make the PR2 robust. However, a robot with the PR2's capabilities can **never** be absolutely safe. The PR2 user's safety, as well as the safety of others, critically depends on the user's constant attention. The PR2 user must be aware of the potential dangers, and learn to anticipate and prevent problems.

2.3 General Usage Guidelines

While many guidelines for the safe use of a robot stem from common sense, a basic set is listed here. It is fundamental to follow these guidelines but note that these guidelines alone do not guarantee safety, only reduce risk.

- Every organization that uses a PR2 must appoint a **Safety Officer.**
  - The Safety Officer's contact information should be known by everyone who uses the PR2 (including designers, developers, programmers, and end-users).
  - Details of the Safety Officer's roles and responsibilities are described in section 2.4.
- **Before operating** or working with the PR2 every user must:
  - View the safety video.
  - Read this user manual, in particular the entirety of Chapter 2 on Safety.
  - Read and understand the latest list of potential hazards, which can be found at safety.willowgarage.com.
  - Know how to contact the organization's Safety Officer.
- **Supervise children, visitors** , and anyone who has not followed the previous guideline. In particular, make sure they:
  - Do not come within range of the PR2 when active.
  - Are aware that the robot could move unexpectedly and is potentially dangerous.

  - Are not alone with the PR2.
  - Do not operate the PR2.

- **Maintain a safe environment.** Safety is not only affected by how a PR2 user operates the robot, but by the environment as well. The PR2 is designed to operate in a laboratory environment.

  - Keep the robot **at least 7 meters from the top of a stairway or any other drop-off.**
  - Make sure the robot has adequate and level space for any expected or unexpected operation.
  - If the robot travels on a ramp, make sure the spine is lowered and the arms are tucked in so that the center of gravity is as low as possible. The slope should not be more than 1:12. Ensure that the robot cannot fall off the edge under any circumstances.
  - Make sure the environment is free of objects that could pose a risk if knocked, hit, or otherwise affected by the PR2.
  - Make sure there are no cables or ropes that could be caught in the covers, wheels, or arms; these could pull other objects over.
  - Make sure no animals are near the robot.
  - Keep fingers, hair, and clothing away from wheels and gears.
  - Be aware of the location of emergency exits and make sure the robot cannot block them.
  - Do not operate the robot outdoors.

- **The PR2 covers are flammable.** Keep the PR2 away from open flames and other heat sources. The PR2 covers may burn if ignited.

  - Never use the PR2 around stoves or ovens.
  - Always be prepared for a fire and have a fire extinguisher close by and ready to use. An ignition source could come from anywhere, including an internal failure of the PR2 system.
  - Use caution to avoid electrical shock.
  - Do not allow the robot to come in contact with liquids (spilled drinks, rain, etc.). This is particularly important when the robot is plugged into an AC outlet.
  - If a robot does get wet, turn off the DC breaker and contact Willow Garage. If it is plugged into an AC outlet, do not touch the robot. Turn off the circuit breaker that controls the outlet.
  - Before removing any covers, the PR2 should be unplugged and the DC breaker should be turned off.

  - Make sure the power cord is in good condition . Cord insulation must be intact with no cracks or deterioration. Both connectors should be undamaged. If the power cord is damaged in any way, it should be discarded and replaced with a new cord from Willow Garage.

- When plugging in the PR2, always attach the power cord to the AC inlet on the robot **before** attaching it to the AC outlet on the wall. Unplug the cord from the wall before unplugging from the robot.
- Use extreme caution whenever the robot's spine is contracting. Objects or body parts could be crushed between the robot's body and base. Also, keep people and objects away from the PR2's joints.
- Make sure the batteries cannot be overheated or punctured. Do not run the robot without its covers. They help protect the batteries during inadvertent collisions.
- The PR2 user should always have immediate access to the **wireless run/stop** box and should stop the robot at the fist sign of a problem. The stop button should always be pressed when examining, inspecting, or maintaining the robot.
- Use **common sense** when operating the robot.
  - Do not allow the robot to grab or hit any person.
  - Do not allow the robot to drive into contact with, or over, any body part.
  - Do not allow the robot to interact with any sharp or dangerous items.
  - Do not allow the robot to operate potentially dangerous appliances (like stoves)or power tools.
- Pay attention to **warning labels** on the robot.
- **Do not remove the covers** of a PR2 or disassemble it in any way without prior and appropriate instruction by Willow Garage.
  - There are high voltages inside the robot.
  - There are a variety of pinch points and other mechanical dangers in the interior of the robot.
  - Counterbalances and springs store significant potential energy which could cause damage if unloaded abruptly.
- Do not modify or remove any part of the **software safety features.**
- The texture projector can trigger headaches or seizures in people sensitive to 50 Hz flashing light.
- When developing software, always consider that some people have adverse reactions to the flicker from the texture projector and only turn it on when necessary.

  - Regularly inspect and maintain anything the robot interacts with. For example, regularly inspect the electrical outlets the robot plugs into.
  - **Anticipate potential problems** and hazards. Always imagine what might happen if the robot malfunctions or behaves in a way different from the desired action. Be vigilant.

- Be aware that many things can go wrong even with a seemingly simple activity. For example, consider what might go wrong while driving the robot with the joystick:
  - The joystick could run out of batteries.
  - The Bluetooth connection between the joystick and the robot could fail.
  - The ROS node that talks to the Bluetooth driver could have been changed or have a bug.
  - The operating system in the robot's computer could hang.
  - The motor controller software node could fail.
  - The motor controller board could fail.
  - The PR2's electrical system could fail.
  - The motor or gear train could have a problem.
  - The wheels could have a problem.
  - Someone else could command the robot to move without the PR2 user's knowledge.
  - The PR2 user could make a mistake. It is intuitive to drive a robot when it is going away from the user, but when it is coming towards the user everything is reversed.

2.4 Safety Program

The possibility of injury can never be completely eliminated when working with a PR2. Therefore, Willow Garage has instituted a safety program to focus continual effort and attention on risk reduction. The program is managed by a Safety Board and has three main goals:

- Anticipation: identifying potential hazards associated with the PR2.
- Mitigation: taking action to reduce the likelihood and severity of hazards.
- Communication: exchanging information about safety within the community.

The foundation of the safety program is an ongoing effort to identify ways a PR2 could cause damage. These are referred to as hazards. For example, a robot could cause damage by running into a person or falling down stairs. The Safety Board maintains a list of all known hazards in a document called PR2 Hazards, which can be found on the safety.willowgarage.com web site. Anyone who works with a PR2 should review this list regularly. If a potential source of damage that is not covered by one of the hazards in the list is discovered, please let Willow Garage know, using one of the contact methods described below.

When a hazard is identified, one or more of the following steps is taken:

- A design change is made to mitigate the potential damage. Examples of these can be found in the document called Design for Risk Reduction on the safety.willowgarage.com web site.
- A guideline is written to let people know how to avoid the potential damage. The list of guidelines can be found in an earlier section of this chapter or in a document called PR2 Usage Guidelines on the safety.willowgarage.com web site.
- A warning decal is added to the robot.

Communication is key to the safety program. There are several sources of information available:

- The web pages at safety.willowgarage.com
- This chapter of the PR2 User Manual
- A Safety Officer for each PR2 site assists with two-way communication.

Anyone who has any concerns about safety or witnesses a safety-related incident should contact the Willow Garage Safety Board.

- By email at safety@willowgarage.com
- By phone at 650-475-9794

CHAPTER 3

What's in the Box

The PR2 ships in 2 packages:

- Robot Crate - Large wooden crate which contains the PR2, accessory kit, base-station computer, and toolkit.
- Calibration Target - Large checkerboard for accurate calibration of stereo cameras.

3.1 The PR2

The PR2 is a robotics research and development platform that allows innovation right out of the box. Please watch the video ”Unpacking the PR2” The video covers un-packing the PR2 and turning on the PR2 while in the crate's base. The video can be found in the tutorials of the PR2's support website.

3.2 The PR2 Accessory Kit

The PR2 accessory kit includes tools and cables for teleoperating, powering, and calibrating the robot. Additionally the sensor fingertips are stored in the accessory kit.

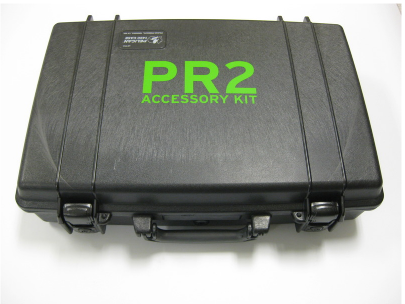

Figure 3.1: The PR2 accessory kit.

- **Wireless Joystick**
  - The PR2 ships with a bluetooth joystick for teleoperating the robot. The bluetooth joystick is a Sony DUALSHOCK3 (Figure 3.2) wireless controller. It can be charged using any standard USB A to mini-B USB cable (one is included in the accessory kit). For more information, see the ps3joy or pr2_teleop package at ros.org.

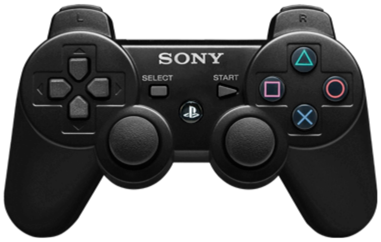

Figure 3.2: The PR2 bluetooth joystick.

- **Wireless run-stop**
  - The PR2 comes with an OMNEX T50 wireless run-stop transmitter. When the red button is pressed, or the unit is out of range, the wireless run-stop transmitter will halt the motors and put the power system in standby mode. Note that this does not completely cut power to the robot.

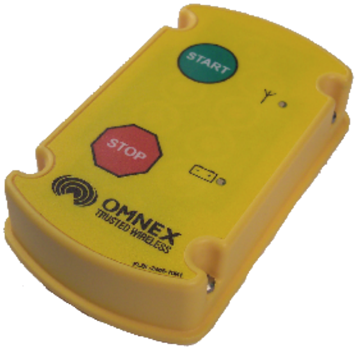

Figure 3.3: The PR2 wireless run-stop.

To start the wireless run-stop, press the green start button (Figure 3.3); a flashing light on the wireless run-stop will appear next to the green start button. While transmitting, the wireless run-stop has a range of approximately 800 ft. The wireless run-stop is powered by four AA batteries; the battery light will flash when the battery charge is low and new batteries are required. It is recommended to always keep spare AA batteries handy.

- **Robot Power Cord and Self Plug-in Robot Power Cord**
  - To recharge the PR2's batteries, it must be plugged into a 120V/220VAC electrical outlet. Use only the provided power cords many power cords and power strips have thinner conductors and cannot safely supply the current required by the PR2.

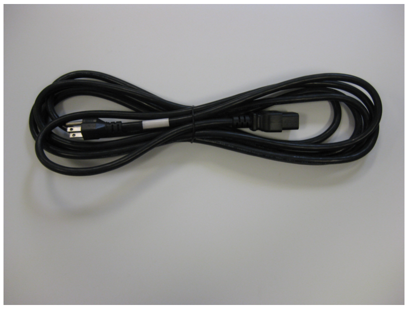

Figure 3.4: The PR2 power cord.

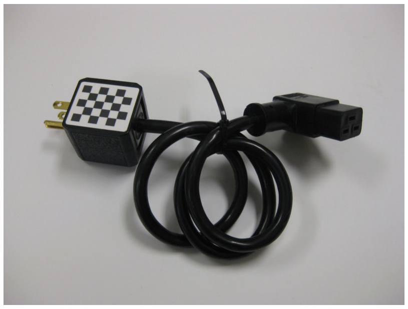

Figure 3.5: The PR2 self plug-in power cord.

  Using inappropriate cables is hazardous and may cause fire. Either cable may be used when manually plugging the robot in to re-charge. When plugging in the PR2, always attach the power cord to the AC inlet on the robot before attaching it to the AC outlet on the wall. Unplug the cord from the wall before unplugging from the robot. When attempting to have the PR2 plug itself in, the shorter self-plug-in cord with attached checkerboard must be used. The self-plug-in cord has magnets on the side opposite the checkerboard which should be used to attach it to the magnetic pad on the base.

- **Sensorized fingertips and boots**
  - The PR2 ships with sensorless fingertips attached. Included in the accessory kit are fingertips which have an integrated 22-element pressure sensor (Figure 3.6). These sensorized tips are easy to damage and are not as robust as the rest of the PR2. Five fingertip sensors are provided (2 for each gripper, plus 1 spare), as well as 20 rubber protective "boots" which prevent the sensor from

coming into direct contact with the environment. When using the fingertip sensors, always have an undamaged boot in good condition installed over the sensor. Continuing to use the fingertip sensors after the rubber boot is damaged greatly increases the chance that the sensors themselves get damaged. Replacement sensor tips will be available for purchase from Willow Garage. See the support site at pr2support.willowgarage.com for more information.

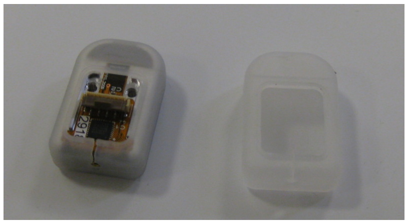

Figure 3.6: Fingertip sensors and replacement boots.

- **Small calibration target**
  - This small hand-held calibration target, which looks like a checkerboard, is used for the PR2 self-calibration. The PR2 holds the checkerboard in one gripper and uses it to calibrate the arms, the head, the cameras, and the tilting laser together. The squares on the board are 25mm in each direction. See the pr2_calibration package for more information.

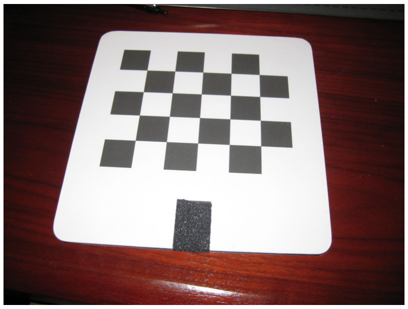

Figure 3.7: Small calibration target.

3.3 The PR2 Toolkit

A toolkit is provided with the PR2. To avoid damaging the robot or personal injury while repairing the PR2 always use these tools and follow the Repair and Replacement procedures at pr2support.willowgarage.com when performing any hardware maintenance on the robot.

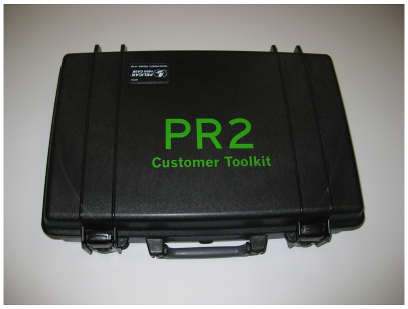

Figure 3.8: The PR2 Toolkit.

3.4 Base-Station Computer

The base-station computer ships without a monitor, keyboard, or mouse. See Chapter 14 on computers for more information on how to configure the base-station.

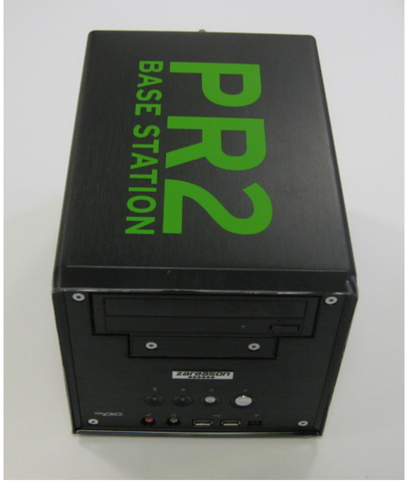

Figure 3.9: The PR2 base-station computer.

3.5 Large Calibration Target

The large checkerboard calibration target, which ships with the PR2, is 1 inch thick and approximately 3 feet square. This is the recommended calibration target to use for calibrating the intrinsics of the stereo cameras on the robot. The PR2 ships with stereo cameras already calibrated, but re-calibration may be needed after shipping, and occasionally as vibration and thermal effects change the parameters over time. The squares on the board are 108mm in each direction. Since the flatness of the board is critical to the accuracy of the calibration, please store this where it will not warp.

CHAPTER 4

Mechanism Terminology

The PR2 kinematics are talked about using the concepts of joints, links, frames, actuators, and transmissions. Each of these is defined and represented in code, as well as having a unique name. In general, a single joint will have an actuator and a transmission and will connect two links together. A link is a rigid body in the kinematic tree, and will have a coordinate frame, but not all frames are associated with links. "Fixed" joints are allowed, so some rigid bodies that could be considered one link are considered to be two links rigidly joined together. This is done mostly at locations where there is a physical interface between components.

4.1 Link

A link is considered to be a rigid body within the robot. It generally has geometry to define visual and collision-space representations, has inertial properties, and has a name. The links for the PR2 are defined in the URDF description that can be found in the pr2_description package.

4.2 Frame

Frames are geometric entities that represent coordinate frames in space. Every link has an associated frame, but frames are also used to represent other things, such as the optical frame of a camera, the origin of global map, or the location of a detected object. Frames are always defined relative to one another, and relationships and transformations between them are tracked using the tf package.

4.3 Joint

A joint is considered to be part of the robot, and define the relationship between links. The joints for the PR2 are defined in the URDF description that can be found in the pr2_description package. The PR2 has mostly rotational joints, but the torso is translational, and there are also a number of "fixed" joints used to represent the notion that two links are part of the same rigid body. Rotational and translational joints are generally represented in the same way in the system, and are referred to as joint "effort" instead of force or torque, as well as using position, and velocity to represent both linear and angular movement.

4.4 Actuator

The motor/encoder units in the PR2 are explicitly modeled as "actuators". The actuator model treats them as torque sources with position measurement. The details of the actuator model (e.g. the motor torque constant or max current), as well as the names of actuators, are stored on the motor controller boards. See the pr2_ethercat package.

4.5 Transmission

Transmissions, also defined in the URDF robot description, are used to represent the mapping between actuators and joints. They map actuator position into joint position (meters or radians), and map from commanded joint efforts (torques or forces) to actuator efforts. In most cases, in the robot this is a linear mapping between one joint and one actuator that just defines a reduction, but the wrist has two motors controlling two joints in a cross-coupled manner, and the gripper the mapping from motor rotation to gripper position is non-linear.

4.6 PR2 Home Pose

In order to describe the PR2 robot pose and joint positions in a consistent manner, a **home pose** of the robot has been defined. At the home pose, all joint angles are considered to be at zero. In the home pose the casters are facing straight ahead, the arms are straight out in front of the robot, the grippers are closed, the tilting laser is level, and the head is centered and level. For most joints the calibration reference point is not at the home pose, or zero position. The URDF of the PR2 contains the offsets between the flag positions and the home positions of the joints.

4.7 PR2 Coordinate System

The coordinate frames for all links of the PR2 are defined to aligned with a world coordinate frame of positive z-axis up, positive x-axis forward, and positive y-axis robot-left when the PR2 is in the home pose. All joint angle conventions are chosen so that from the home position, positive motion of the joint causes positive motion around one of the positive axes of the world coordinate frame.

4.8 Naming Conventions

In general, the names for a link and the associated frame will be similar (e.g. r_forearm_link and r_forearm_frame), and the names for an actuator, a transmission, and the associated joint will be similar (e.g. r_elbow_flex_motor, r_elbow_flex_trans, and r_elbow_flex_joint). For components such as arms and casters that are included in the robot in multiple locations, a short prefix is used to indicate which location is referred.

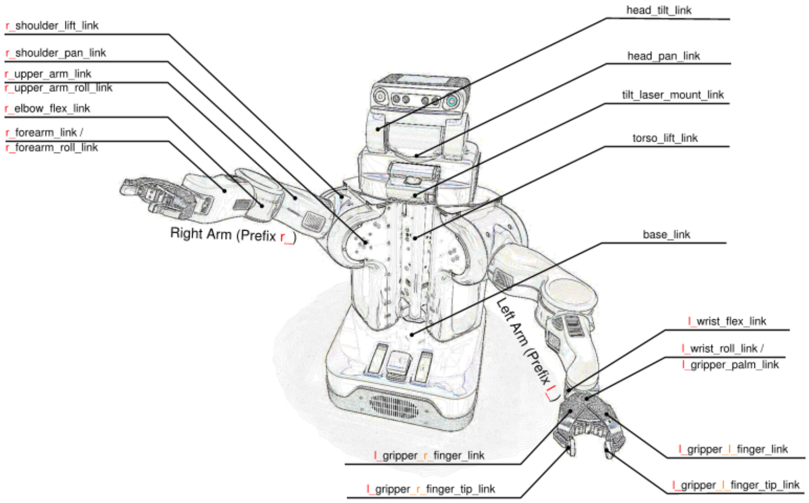

Figure 4.1: The PR2 link naming scheme.

CHAPTER 5

Links and Joints

5.1 Casters

Each robot has four steered and driven casters, one at each corner of the base. The casters are referred to as front right (`fr_`), front left (`fl_`), back right (`br_`), and back left (`bl_`). Each caster consists of a rotation joint `*_caster_rotation_joint`, which rotates or steers the body of the caster, called the `*_caster_rotation_link`, around a vertical axis. Attached to the body of the caster are two parallel driven wheels, whose drive joints are `*_caster_r_wheel_joint` and `*_caster_l_wheel_joint`.

5.2 Base-Body-Spine

The four caster of the robot attach to the base_link of the robot via the `caster_rotation_joint`. Physically the base_link of the robot includes the rear panel with electrical/network connections, computers, and battery system. The torso of the robot is a prismatic joint described by the `torso_lift_joint`, which moves the torso up and down relative to the `base_link`. The `torso_lift_link`, which moves up and down, contains the speaker, and the IMU, as well as serving as the attachment point for both arms and for the head.

5.3 Head

The pan-tilt head which holds the wide and narrow stereo cameras, the 5 mega-pixel camera, and the texture projector is made up of two links the `head_pan_link`, connected to the torso by the `head_pan_joint`, and the `head_tilt_link`, connected to the pan link by the `head_tilt_joint`. The sensors are all attached to the tilt link. The tilting Hokuyo laser range-finder is mounted to the `laser_mount_link`, which is tilted up and down by the `laser_tilt_mount_joint`.

5.4 Arms

The arms of the PR2 attach to the torso_lift_link at the *_shoulder_pan_joint which connects to the *_shoulder_pan_link. The *_shoulder_pan_link represents the counterbalance assembly of the arm. Connected to the *_shoulder_lift_link through the *_shoulder_lift_joint. The *_shoulder_lift_joint moves the arm up and down. The *_upper_arm_roll_joint allows the upper arm of the PR2 to rotate 268 degrees. The *_elbow_flex_joint connects the *_upper_arm_roll_link and the *_elbow_flex_link allowing the arm to bend similar to a human elbow. The *_forearm_roll_joint attaches the *_forearm_roll_link to the *_elbow_flex_link and allows the *_elbow_flex_link to rotate continuously. The *_elbow_flex_link is connected to the *_wrist_flex_link by the *_wrist_flex_joint. The *_wrist_flex_link connects to the *_wrist_roll_link by the *_wrist_roll_joint which can rotate continuously. All together the PR2 arm has 7 DOF.

5.5 Grippers

The gripper consists of a central palm link called the *_gripper_palm_link which is rigidly fixed to the *_wrist_roll_link and has a single actuated degree of freedom. The motor in the gripper drives the four joints of the gripper together to result in parallel motion of the fingertips. The parallel distance between the gripper tips is treated as a translational joint, called *_gripper_joint that can be controlled. The four rotational joints in the hand which are constrained to move together are also modeled as joints which can be observed but not controlled. The gripper also contains a 3 DOF accelerometer and an LED which can be turned on and off to assist in calibration or finding the location of the gripper in camera images.

5.6 Additional Information

A CAD approximated kinematic and inertial description of the PR2 can be found in the pr2_description package. This package uses a robot specific Extensible Markup Language (XML) and Document Object Model (DOM) representation called Unified Robot Description Format (URDF), which can be converted or exported to other formats such as COLLADA.

CHAPTER 6

Drivetrain Overview

The joints in the PR2 are all actuated with brushed electric motors, and all position sensing is done with optical incremental quadrature encoders. Except where noted, each joint has an optical sensor (interrupt or reflective) which is used at startup to identify a home position. Once the home position is identified, the incremental encoders are used to track motion. All drivetrains in the robot have positive-engagement drives (gears or belts), and the motor controllers monitor for encoder errors. The repeatability of the calibration sensors may cause minor differences in position sensing between runs.

Velocity estimates are made by differentiating the position sensors at 1ms, so the quantization of the encoders produces noise in the velocity signal, which is significant at low speeds.

Torque estimates are made by measuring the commanded current of the motor and using the values of the torque constant from the data-sheet of the motor together with the model of the transmission. These transmission models assume 100% efficiency, and accuracy of torque estimates and commands are limited by the friction in the drivetrain, so joint torque accuracy should be characterized where one is relying on it.

6.1 Drivetrains

The three joints in each caster have basically the same drivetrain, which is a Maxon amax-32 motor with a gear-head and a continuous belt output. Position sensing is done via an encoder on the back of the motor. The drivetrain is efficient enough to be easily backdriveable. All three motors work together to rotate the caster. The rotation motor drives the rotation and the two wheels drive in opposite directions to provide additional forces. To drive, the two wheel motors rotate in the same direction and the rotation motor helps to keep them aligned in the proper direction.

6.2 Drivetrain

The torso is driven up and down on a linear rail by a combination of a gas spring, which supports most of the weight of the torso, and a lead-screw which is driven via an Maxon RE40 motor with 113:1 gear-head connected to a continuous belt drive with a 5:1 reduction. As a result of the force from the gas-spring not being constant over the range of travel, the lead-screw may be loaded up, down, or neutrally as different points of travel of the torso. Position sensing of torso lift is done via optical encoder on the back of the motor, so any backlash in the lead-screw, belt, or gear-head is not sensed. The torso joint has low efficiency, so commanding large forces (in the range of 1000 newtons) will be required to move the torso. The maximum speed of the torso lift joint is limited by the thermal rating of the lead-screw nut. Driving the torso lift link faster than the velocity limit will cause permanent damage to the lead-screw assembly. Similarly, driving at the safety limit but with excessive load because of the large reduction, the torso lift link is capable of exerting high forces.

6.3 Drivetrains

The drivetrains for the head pan, head tilt, and laser tilt are all fairly similar. Since precise pointing of sensors is important, all three of these joints have 10,000 count encoders mounted in the drivetrain after the major sources of potential backlash (gears). The laser tilt axis is driven directly by the motor, through the encoder, and then via a belt-drive. The head pan and tilt are both driven from the motor through a gear-set (planetary gear-head for the tilt joint and external spur gears for the pan joint). The encoder is mounted on the shaft after the gear-set, and then a belt drive is used for the final stage reduction. These drivetrains are all high-efficiency and capable of very high velocities. The software velocity and position limits on these joints are important to prevent hard collisions with the joint stops, which will cause damage to the hardware (especially to the sensors). If the mass of the head changes significantly, the pan and tilt link gains may need to be re-tuned.

6.4 Counterbalance

Although not an actuated drivetrain, the arm counterbalance is one of the more complicated mechanisms on the PR2. Inside of the shoulder turret there are two springs. These springs are attached to belts which run over cams and provide a uniform downward force.

In the upper arm there is a four-bar linkage which transmits the torque through to the shoulder lift joint, upper-arm roll joint, and elbow flex joint. As the arm moves through its configuration space the linkage changes the amount of leverage with which the counterbalance is pulling which compensates for the effect of the mass of the arm in different configurations.

6.5 Drivetrains

The first four degrees of freedom in the arm (shoulder pan, shoulder lift, upper-arm roll, and elbow flex) are all driven with Maxon RE40 motors that have single-stage planetary gear-heads which output to a toothed pulley that runs on a belt. These belts all have terminations at either end of a capstan on the moving link, which is how the torque is transmitted. The terminations are also how these belts are tensioned. Proper belt tensioning is important to prevent backlash (which can come from extra slack in the belt), and excess wear (which can come from a belt which is too tight, too loose, or just misaligned). The position is measured via an optical encoder on the back of the motor, so the backlash of the gear-head and the stretch of the belt both contribute to the error in the sensed position. Velocity limits are important on these joints to prevent motor damage from running over speed, as well as to prevent rapid deceleration which can damage the motor, gear-head, and belt.

All of these drivetrains have fairly high efficiency, which means they can be back-driven. This is made possible by the low force requirements on those joints due to the arms counterbalance. The counterbalance uses a system of springs and linkages to compensate for the effect of gravity on the upper arm and forearm links. The compensation should be effective throughout the range of motion of the first four links of the arm.

6.6 Drivetrain

The forearm roll joint is driven by a motor in the upper-arm with a planetary gear-head and external spur gear. Position is measured via an optical encoder on the back of the motor. Because of the higher hear reduction, the efficiency and backdriveability of this joint is not as good as that of the first four joints of the arm.

The wrist flex and roll use a differential drive between two motors. If the motors are turned in the same direction the wrist rolls, if they are turned opposite the wrist flexes. The differential bevel gears are attached after a standard encoder, motor, and gear-head assembly.

6.7 Drivetrain

The gripper consists of two fingers with two joints each. Within each finger, a four-bar linkage constrains the two joints to move together so that the fingertips remain parallel. The two fingers are constrained together with a sector gear that meshes between the first phalanges of the two fingers. The motion of the fingers is driven by a motor which drives a lead-screw in the palm of the hand. This lead-screw is embedded in the four-bar linkage so that the rotation of the motor has a nonlinear relationship to the movement of the joints. The gripper can be back-driven when significant force is applied. Unlike all other non-continuous joints in the system, the gripper has no calibration flag. Initial homing is done by closing the gripper until it gets blocked and then treating that position as a zero point. This means that the gripper is unable to detect if it has mis-calibrated on startup.

CHAPTER 7

Motion Control

PR2 motion control describes the electrical and mechanical hardware in combination with software to produce the desired joint effort, position, or velocity control. Figure 7.1 shows the hardware and software layer interaction, as well as the realtime and non-realtime processes that occur in the motion control loop.

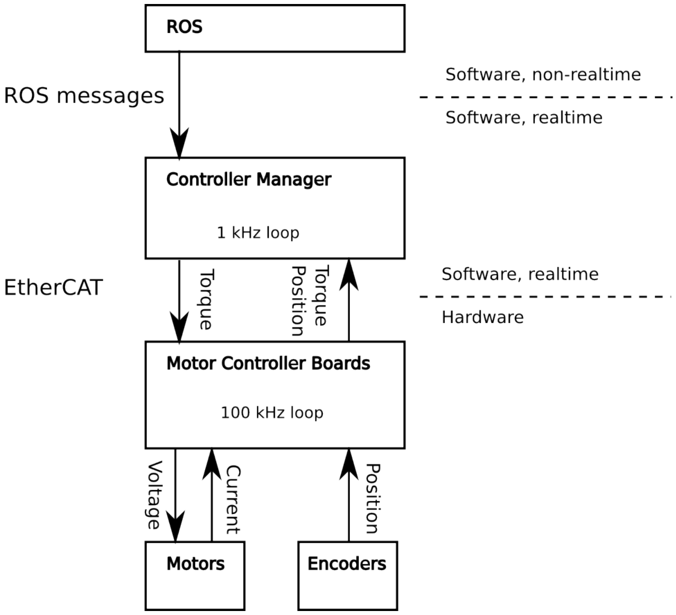

Figure 7.1: The PR2 motion control layout.

7.1 Motor Controller Boards

Each motor/encoder on the PR2 has its dedicated Motor Controller Board (MCB). The MCB detects and counts transitions in the encoder signal, measures the motor current, and commands the voltage going to the motor. Each MCB runs a PI-control loop to control the motor current to a desired value, by commanding the motor voltage. This control loop is closed at 100kHz on an FPGA on the MCB. A shared EtherCAT realtime Ethernet link with the first computer allows all MCBs in the PR2 to communicate with the main computers with deterministic timing.

Each MCB has the following inputs and outputs:

- **Encoder Input**
  - Each MCB can read quadrature incremental encoder input. In proper operation, this should never skip or drift.
- **Power Input**
  - Each MCB is connected to the raw battery-power bus, and using the circuit breakers there can be a switch between ”enabled”, which provides 44-80VDC to the MCB and allows it to drive the motor, ”standby”, which provides 18 VDC to the MCB and allows it to continue to communicate and read sensors but does not allow driving the motors, and ”disabled”, which turns off power to the MCB altogether.
- **Digital Input**
  - Each MCB has a digital input which is used for detecting the flags at the calibration positions on power-up. The MCB internally latches the encoder counts at the time that the digital input transitions from high to low and from low to high, which accurate detection of calibration flag positions.
- **Digital Output**
  - Each MCB has a digital output. On the gripper, this is used to control the LED mounted in the hand. Elsewhere in the robot, this is used to generate synchronization pulses which allows synchronizing the cameras, the projector, and the motion control systems together.
- **Motor Output**
  - Each MCB can drive a single brushed DC motor. The MCB drives the output with an H-bridge and uses a current-measurement to do closed-loop current control on the motor.
- **SPI**
  - There are two SPI ports which are only available on the gripper MCB’s and which are used to connect to the fingertip sensors.

7.2 Realtime Loop

The realtime loop for the PR2 is a single process, which can be found in the pr2_ethercat package. At runtime, the node is called the realtime loop.

This process uses the RT PREEMPT extensions to the Linux kernel to run as a hard realtime process, with a cycle time of 1ms, and performs both controller update computations and communication over EtherCAT with the MCBs. Non-real-time threads within the same process are used to communicate over ROS to publish diagnostics, configure controllers, and otherwise control the behavior of the PR2's motion-control system.

7.3 Controller Manager

Within the realtime loop, the controller update is dealt with by a ROS component which contains both realtime and non-realtime functionality, called the controller manager, see the pr2_controller_manager for more information. Controller manager has an update step which is called by the realtime loop at 1khz which receives measured motor torques and positions and sends desired motor torques for the next cycle.

The controller manager includes the joint-limit safety code that enforces position, velocity, and torque limits which are necessary to keep the PR2 from damaging its hardware. In addition, it has a dynamic plugin loading mechanism that allows shared libraries containing realtime controllers to be loaded and executed inside the 1kHz realtime loop.

CHAPTER 8

Mechanical Specification

Do not operate the PR2 before reviewing the mechanical specifications listed below. Please contact Willow Garage Support at safety@willowgarage.com regarding any questions about the PR2's specifications.

8.1 Environmental

The PR2 is an indoor, household, robot. Operating outside this type of environment could cause damage to the PR2, and injury or death to operators.

- **Drive Surface**
  - The drive surface of the PR2 must be capable of supporting the entire weight of the PR2, about 450 pounds (220 kgs). If the surface is too soft, the PR2 can get stuck and fail to drive. A commercial carpet or tile is recommended.
- **Incline Surface**
  - The PR2 is ready for ADA-compliant ramps, which are at 1/12 slope. Ramps that are steeper than a 1/12 slope are unsafe and may be a tip-over hazard.
- **Water**
  - The PR2 has not been tested for any type of contact with water or any other liquid. Under no circumstances should the PR2 come in contact with water from rain, mist, ground water (puddles) and any other liquid. Water contact can cause damage to the electrical circuitry and the mechanism.
- **Temperature and Humidity**
  - Temperature testing of the PR2 has allowed the unit to run between 15C and 35C. Towards the upper end of this range, above 30C, decreased performance of the batteries may be experienced. Temperatures outside of this range can

cause malfunctions in the PR2 power system and instruments. The PR2 has been tested in high humidity, but under no circumstances should condensation be allowed to form on the mechanism.

- Keep the PR2 away from open flames and other heat sources. The PR2 covers may burn if ignited. Never use the PR2 around stoves or ovens. Always be prepared for a fire and have a fire extinguisher close by and ready to use. An ignition source could come from anywhere, including an internal failure of the PR2 system.

- **Other Environmental Specs**
  - UV exposure should be minimized. UV radiation can damage the PR2's skin.
  - Dust and dirt can clog air filters Forces and torques.

8.2 Forces and Torques

Joint position, velocity, and force limits are implemented in the PR2's URDF file, in the "/etc/ros/distro/urdf/robot.xml" file on the PR2. These joint limits control the range of travel of the mechanism, the allowable velocity to prevent over-travel. These limits are enforced by pr2_controller_manager, and are designed to prevent poorly commanded control efforts from damaging the PR2 and harming operators.

The limits below are from the PR2 URDF file. If a velocity or torque limit is not specified, no value is enforced by pr2_controller_manager.

Joint	Velocity (rad/s or m/s)	Torque (Nm or N)
∗_caster_rotation_joint	-	-
∗_caster_wheel_∗_joint	-	-
torso_lift_joint	0.013	10000
laser_tilt_joint	10.00	0.65
head_pan_joint	6.00	2.65
head_tilt_joint	5.00	15.00
∗_shoulder_pan_joint	2.10	30.00
∗_shoulder_lift_joint	2.10	30.00
∗_upper_arm_roll_joint	3.27	30.00
∗_elbow_flex_joint	3.30	30.00
∗_forearm_roll_joint	3.60	30.00
∗_wrist_flex_joint	3.10	10.00
∗_wrist_roll_joint	3.60	10.00
∗_gripper_joint	0.20	1000

The PR2 motor controller boards (MCB's) will not allow a current command greater than the maximum continuous current specified for the joint's actuator. This means that maximum joint effort may be lower than the maximum effort specified above. Below are the actuators for each joint (Maxon part number), and their maximum allowable commanded current.

Joint	Motor	Power (W)	Max Current
∗_caster_rotation_joint	236672	20	0.655
∗_caster_wheel_∗_joint	236672	20	0.655
torso_lift_joint	148877	150	3.12
laser_tilt_joint	310009	60	1.72
head_pan_joint	310009	60	1.72
head_tilt_joint	310009	60	1.72
∗_shoulder_pan_joint	148877	150	3.12
∗_shoulder_lift_joint	148877	150	3.12
∗_upper_arm_roll_joint	148877	150	3.12
∗_elbow_flex_joint	148877	150	3.12
∗_forearm_roll_joint	310009	60	1.72
∗_wrist_flex_joint	310009	60	1.72
∗_wrist_roll_joint	310009	60	1.72
∗_gripper_joint	222057	11	0.204

More information about each actuator may be found in Maxon data-sheets.

8.3 Joint Limits and Types

The position limits for the PR2 are specified below. These ”hard limits” are the maximum travel for the mechanism.

Joint	Type	Limit (+)	Limit (-)
∗_caster_rotation_joint	continuous	-	-
∗_caster_wheel_∗_joint	continuous	-	-
torso_lift_joint	prismatic	310 mm	0 mm
laser_tilt_joint	revolute	85°	-45°
head_pan_joint	revolute	168°	-168°
head_tilt_joint	revolute	60°	-30°
r_shoulder_pan_joint	revolute	40°	-130°
l_shoulder_pan_joint	revolute	130°	-40°
∗_shoulder_lift_joint	revolute	80°	-30°
r_upper_arm_roll_joint	revolute	44°	-224°
l_upper_arm_roll_joint	revolute	224°	-44°
∗_elbow_flex_joint	revolute	133°	0°
∗_forearm_roll_joint	continuous	-	-
∗_wrist_flex_joint	revolute	130°	0°
∗_wrist_roll_joint	continuous	-	-
∗_gripper_joint	prismatic	86 mm	0 mm

8.4 Modifying Joint Limits

On the PR2, "soft limits" stop the joints from reaching the full range of motion to prevent damage to the mechanism. These soft limits, similar to a virtual spring, are specified in the robot's URDF file. For an explanation of their implementation, see safety_limits in the pr2_controller_manager package.

The soft limits have been validated by Willow Garage, and are required for safe operation of the PR2. Under no circumstances should these limits be increased without prior written authorization by Willow Garage Safety. Unauthorized and unvalidated modification of these limits could cause mechanism damage and increase the risk of injury or death to PR2 operators.

Mechanism damage resulting from changing these limits can include:

- Thermal damage to motors from exceeding maximum current specifications.
- Damage to drive-trains resulting from collision with end-stops.
- Premature wear and damage to motor, gear-head, or drive-train resulting from operating over the components' rated speed.
- Premature wear of slip-rings or other electrical connections from operating over the components' rated speed.
- Vibration damage to sensors or other components resulting from excessive velocities or collisions with joint-stops.

CHAPTER 9

Sensor Overview

The PR2 has a variety of sensors on its body: Insert Figure Here Figure 3.12: Sensor Modularity Diagram

9.1 Base Laser

The base laser is a Hokuyo Top-URG (UTM-30LX) scanning range finder. This laser has a 30 m and a 270 degrees scanning range. For more information, see the hokuyo_node package at ros.org.

9.2 Tilting Laser

In addition to the base laser, the PR2 has a Hokuyo Top-URG (UTM-30LX) mounted on a tilting platform located just below the pan-tilt head. The tilting platform can sweep the scanning laser through 135 degrees (+90 degrees and -45 degrees from level) and can be controlled using the default laser_tilt_controller. For more information, see the hokuyo_node and pr2_default_controllers packages at ros.org.

9.3 Head Cameras

The PR2 pan-tilt head has three cameras and a textured light projector:

- **Wide Stereo Camera**
  - The wide stereo camera is part of the dual stereo pari and is a 100Mb color ethernet camera. The wide stereo uses the Aptina MT9V032C12STC imager chip and has a maximum resolution of 752 x 480 pixels at 15 fps. The camera has a field of view (FOV) of approximately 90 degrees and a 2.5mmm F2.5 Marshall V-4402.5-2.5-HR lens. For more information, see the wge100_camera package at ros.org.

- **Narrow Stereo Camera**
  - The narrow stereo camera is part of the dual stereo pair and is a 100Mb monochrome ethernet camera. The narrow stereo uses the Aptina MT9V032C12STM imager chip and has a maximum resolution of 752 x 480 pixels at 15 fps. The camera has a FOV of approximately 55 degrees and a 5.6mm F2.0 Marshall V-4402.5-2.5-HR lens. For more information, see the wge100_camera package at ros.org.
- **Gigabit Ethernet Camera**
  - The PR2 has a gigabit ethernet camera located to the left of the dual stereo pair on the pan-tilt head. The gigabit ethernet camera is a Prosilica GC2450C, which uses the Sony ICX-625AQ imager chip and has a maximum resolution of 2448 x 2050 pixels at 15 fps. Additionally, the gigabit ethernet camera has a 8mm F1.4-F16 Kowa LM8JC lens. For more information, see the prosilica_camera package at ros.org.
- **Textured Light Projector**
  - The PR2 has a textured light projector located to the (PR2's) left of the dual stereo pair on the pan-tilt head. The projector has a FOV of approximately 55 degrees and a 5.6mm F2.0 Kowa LM12JC lens. For more information, see the pr2_camera_synchronizer package at ros.org. Note: The projector will only work if the motors are enabled and the runstop is not hit.

9.4 Gripper Sensors

- **Accelerometer**
  - The gripper of the PR2 is equipped with a Bosch BMA150 digital triaxal accelerometer. The measurement range (2g, 4g, or 8g) and bandwidth (25Hz-1500Hz) of the accelerometer can be selected in software. For more information, see the ethercat_hardware package at ros.org.
- **Fingertip Pressure Sensors**
  - The default fingertips of the PR2 are sturdy aluminum blocks with non-slip rubber covers for added friction and compliance is grasping. However, the aluminum tips can be swapped out for an (included) set of RoboTouch tactile sensing pads made by Pressure Profile Systems, each with 22 tactile sensing elements: fifteen in a 5x3 array on the front surface, two on top, two on each side, and one in back, near the top. Each tactile element has a pressure range of 0-30 psi (0-205 kPa) and sensitivity of 0.1 psi (0.7 kPa). The sensors connect to the robot via an SPI ribbon cable and two screws, and have a maximum scan rate of 35 Hz. The tactile sensors are highly fragile when not protected by the rubber covering, and so care should be taken not to damage the rubber covering. For more information, see the fingertips_pressure package at ros.org.

- **Calibration LED**

9.5 Inertial Measurement Unit

The PR2 has an inertial measurement unit (IMU) located next to the tilting laser. The IMU is a MicroStrain Inertial-Link 3DM-GX2 which has an accelerometer range or 5g and a gyro range of 300 degrees/s. For more information, see the microstrain_3dmgx2_imu package at ros.org.

9.6 Speaker

The PR2 has one Logitech V20 notebook speaker located under the pan-tilt head next to the tilting laser. For more information, see the sound_play package at ros.org.

CHAPTER 10

Power System

Place holder for the power system

10.1 Subsection

CHAPTER 11

Computer Overview

PR2 has two computers, each has 24 Gb of RAM, 2 quad-core Nehalem processors, and both a 2.5 inch internal hard-drive and an externally removable 3.5 inch drive.

Additionally, the PR2 ships with a basestation computer which facilitates seamless communication with the PR2 when transitioning between wired and wireless networks, and additionally helps in a number of maintenance tasks.

11.1 Computer 1 (c1)

Computer 1 (c1) is physically located on the right side of the robot. It is referred to as the master computer because it serves a number of key roles for the computer infrastructure:

- c1 stores the operating system of both computers. C2 cannot boot unless c1 has booted first.
- c1 is connected to the PR2 ethercat network, and is the only computer that can perform motor control.
- c1 provides routing for the rest of the robot when it is plugged in via the WAN port.
- c1 provides routing for the rest of the robot when connected to another network via an open VPN tunnel

c1's PCI slot is used for a 4-port ethernet card, giving it a total of 6 ethernet ports. These are:

- lan0 - lan3: connection to internal robot network
- wan0: connected directly to WAN port on back of robot
- ecat0: connected to robot ethercat network

11.2 Computer 2 (c2)

Computer 2 (c2) is physically located on the left side of the robot. It is sometimes referred to as the slave computer because it netboots from c1.

c2 only has 2 ethernet ports, lan0 and lan1, both of which are connected to the internal robot network.

11.3 Basestation

The basestation is a Zareason xPC. It is a dedicated point of contact through which network traffic to the robot can be routed. It has 2 ethernet ports, wan0, on the motherboard, is the primary ethernet port. It is intended to be plugged into your building network. lan0, on the pci card is a dedicated port for servicing the robot. When necessary, this port should be plugged directly into the robot service port.

When configured properly, it is important that the basestation is visible on UDP port 1194 (or other user-configurable port) both via the wired building network as well as via the building wireless network. This will likely require assistance from the local network administrator.

11.4 Default Password Reference

There are many different devices which make up the PR2. Many of these have passwords which should be changed for security purposes. The default logins and passwords are:

- Basestation (10.68.0.6). Standard unix logins:
  - root - willow
  - pr2admin - willow
- C1 (10.68.0.1). Standard unix logins:
  - root - willow
  - pr2admin - willow
- C1 (10.68.0.1). Web-interface login:
  - admin - admin
  - guest - guest
- C1 ESMS (10.68.0.91
  - root - changeme
- C2 ESMS (10.68.0.92)
  - root - changeme

- Wireless Router (10.68.0.5)
  - root - willow
- Wireless Access Point (10.68.0.250)
  - root - willow

CHAPTER 12

Networking

The majority of communication between components onboard the PR2 happens via an onboard ethernet network, referred to as the ”robot internal network”. This onboard network can be accessed directly via either wired connection (through the service port), or wireless connection (through the WAP). Additionally, the robot can be accessed via the basestation through a VPN tunnel.

12.1 Network Segments

There are a couple of different important network segments, outlined here, and discussed in more detail in the section 12.2 below.

10.68.0.0/24: Primary Robot Internal Network. The 10.68.0.0 subnet is the primary network used internally by the computers on the robot. Both computers and the basestation have addresses on this subnet. Additionally many of the ethernet-based devices, such as the power board, are given addresses on this subnet. The robot Service Port, and the cradle-point ctr350 Wireless Access Point are directly connected to this network, allowing a user to easily put their laptop on the robot internal network. c1, if booted, will give out IP addresses via DHCP in the range 10.68.0.100-10.68.0.199. Important addresses on this network:

- 10.68.0.1 - c1 (lan0)
- 10.68.0.2 - c2 (lan0)
- 10.68.0.5 - wifi-router
- 10.68.0.6 - basestation (lan0)
- 10.68.0.91 - c1-esms
- 10.68.0.92 - c2-esms
- 10.68.0.250 - wap

10.69.0.0/24: Secondary Robot Internal Network. The 10.69.0.0 network is a second internal subnet. The cameras are given addresses on this subnet to partition traffic across the computer's two network interfaces of the computers. This way, heavy network utilization by the cameras is less likely to interfere with more critical computer functions such as NFS.

- 10.69.0.11 - c1 (lan1)
- 10.69.0.12 - c2 (lan1)

10.68.X.0/24: Robot VPN Network. The primarily role of the basestation is to function as a VPN server for the robot. Each robot can be given a unique VPN subnet to facilitate the operation of multiple robots using a single VPN server. The basestation can be configured to automatically forward relevant traffic from the building network into the VPN network, hiding this from the end-user. However, if greater security is desired, the basestation can instead be configured to require users to be assigned a VPN key to access the PR2's VPN network. Important addresses on this network:

- 10.68.X.1 - cl (tun0)
- 10.68.X.2 - c2 (via cl:tun0)
- 10.68.255.1 - basestation (tun0)

12.2 Network Explanation

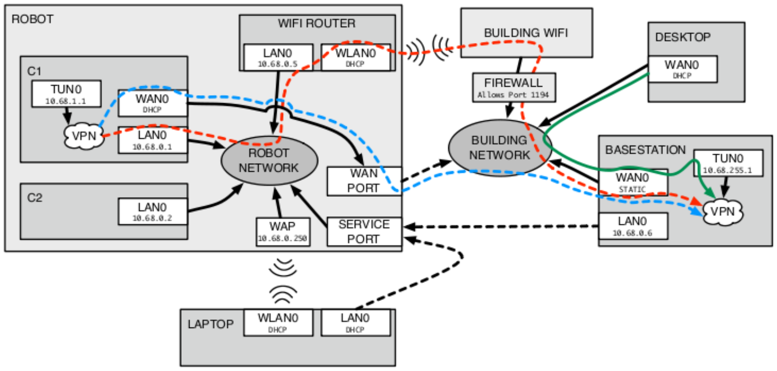

Figure 12.1: The PR2 network diagram.

The above figure depicts the interaction between the robot and building networks. The dotted lines show paths which may or may not be connected at any given point in time. That is, the basestation or a laptop may not always be plugged into the service port, and the robot WAN port may not always be plugged into the building network. The colored lines depict pathways that different network traffic follows.

The main thing of importance to note is that there are three ways to talk to the robot: direct wire, direct wireless, or VPN. Additionally, the VPN traffic itself can be routed in one of two ways, either via a wired or wireless path. **Note:** for clarity, all interfaces connected to the building network as WAN, and interfaces connected to the robot network as LAN have been labeled, but this may not be the case for laptops or desktops on the actual network, where the usual default interface name is typically "eth0"

- **Direct wired connection**
  - One possibility is to connect a laptop, or the basestation lan0 port to the PR2 service port. This simply puts the laptop, or the basestation directly on the PR2's internal network. The basestation is always configured to use the IP address 10.68.0.6, whereas the laptop will likely be configured to use DHCP. When plugged into the service port, the PR2 user can talk to the PR2's LAN ports directly on the 10.68.0.0/24 subnet. This system will work even if the VPN system is not functioning. This is typically used for reinstalling the robot, or debugging other networking problems.
- **Direct wireless connection**
  - Another possibility is to connect a laptop directly to the PR2's WAP. This is essentially identical to plugging into the service port. The PR2 computers can be reached directly on the 10.68.0.0/24 subnet.
- **VPN connection**
  - When not plugged int the service port, the basestation and PR2 communicate over VPN. The basestation acts as the VPN server, and is always reachable at 10.68.255.1. However, there are two mechanism through which the VPN traffic may be routed.
  - **Wired VPN connection**
    - ∗ The wired VPN connection is depicted in blue. This only occurs when the robot Wan port is plugged into the building network. Note that wan0 on c1 goes directly to the robot WAN port. When this is plugged into the building network, c1 acquires an IP address via DHCP. At that point c1 initiates a connection to the basestation using a specified static IP address. Once this connection is established, all VPN traffic is routed through this pathway.
  - **Wireless VPN connection**
    - ∗ The wireless VPN connection is depicted in red. If the WAN port is not plugged in, the robot falls back on its wireless connection. A secondary route instead send traffic through the wifi-router, which must be configured to connect to the building wireless network. In many cases, the wireless network is located outside the firewall, so it is important that the

static IP of the basestation resolves properly and is allowed through the firewall on port 1194.

Regardless of how the robot is connected to the basestation, a desktop situated on the building network will always talk to the robot in the same way. Traffic from the desktop, depicted in green, is actually routed through the basestation and into the VPN tunnel. Once in the tunnel, which pathway the robot is using is abstracted and the traffic seamlessly emerges from the other side of the tunnel.

12.3 Service Port

The PR2 service port is the bottom ethernet port on the back panel of the robot. It connects directly to the PR2 internal network, allowing a connection to the computers directly. **DO NOT PLUG THIS PORT INTO THE BUILDING NETWORK** . c1 serves DHCP for this network and if it conflicts with another DHCP server this will most likely cause problems both on the building network and on the PR2 network depending on which DHCP server takes precedence.

12.4 WAN Port

The PR2 WAN port is the top ethernet port on the back panel of the PR2. This connects directly to wan0 on c1. This port is intended to be plugged into the building network. The robot will attempt to acquire an IP address via DHCP, and then attempt to contact the basestation at a known IP address.

12.5 Wireless Access Point

The PR2 comes configured with a cradle-point ctr350 configured as a Wireless Access Point (WAP). The ESSID of this network defaults to ”ROBOTNAME_LAN” and allows direct access to the Robot Internal Network. The WAP can be configured by plugging into the PR2's service port and going to the IP address: 10.68.0.250. The default login and password are ”root” and ”willow” respectively.

12.6 Wireless Router

The PR2 has a Linksys WRT610N dual-N band wireless router. This router can be configured to connect to the building wireless network. In the absence of a WAN connection, the robot will attempt to contact the basestation through this wireless router instead. The wireless router can be configured by plugging into the robot service port and going to the IP address: 10.68.0.5. The default login and password are ”root” and ”willow” respectively.

CHAPTER 13

Computer Configuration

The operating system running on the PR2 computers is an extended version of Ubuntu 9.04 (Jaunty Jackalope) or 10.04 (Lucid Lynx). It depends on a number of additional packages for system configuration, but should otherwise be familiar for Ubuntu users. If there is a computer problem not covered by the PR2 documentation, the Ubuntu Documentation is the next place to look.

13.1 BIOS

The BIOS configuration is slightly different between c1 and c2 to allow them to boot differently. c1 defaults to booting off of the internal disk, while c2 defaults to netbooting off the lan0 interface. Early in the boot process, c1 is reconfigured to try netbooting off of the the lan1 interface before booting off the disk (using lan1 rather than lan0 enables the basestation to distinguish the 2 computers). On a proper shutdown, c1 is switched back booting off the disk, but in the event of a hard power failure, the default BIOS state leaves c1 netbooting. This is convenient for reinstallation or recovery.

13.2 Networking

To keep connections behaving sensibly when transitioning the VPN connection to the basestation between wired and wireless, the robot is configured to route ALL traffic that is not on the PR2 internal network out through the basestation. Depending on if the robot has been installed as Jaunty or Lucid, the mechanism that allows this is slightly different, and discussed below.

In both OS versions, iptables is configured using configuration files in `/etc/iptables.d/`. These rules are responsible for providing fire-walling on the robot wan0 interface, turning on masquerading, and NATing incoming VPN traffic into the robot local network.

13.2.1 Jaunty Networking

Any traffic not routed via the basestation is sent to a blackhole to keep connections from getting broken. This means that if VPN is not connected the PR2 will not be able to see the outside world even though it might appear that the network is configured appropriately. This is handled using an ip rule configuration. Most of these rules are added by the init-script: `/etc/init.d/iprules/sh`.

The current rule configuration can be viewed using the command ip rule.

```
$ ip rule
0: from all lookup local
1000: from all to 10.68.0.0/24 look up main
1001: from all to 10.0.2.239 look up main
1002: from all to 10.68.1.0/24 look up main
2000: from all look up VPN
3000: from all blackhole
32766: from all lookup main
32767: from all loookup default
```

The two important rules are priority 2000 and 3000. The priority 2000 rule routes all traffic into the VPN routing table:

```
$ ip route list table VPN
default via 10.68.1.6 dev tun0 metric 1
```

The priority 3000 rule is what sends things to the blackhole. By disabling the priority 3000 rule, traffic which cannot be routed through the basestation, such as when the VPN tunnel is down, will simply go to the main routing table.

To remove the priority 3000 rule, run:

```
$ sudo ip rule del priority 3000
```

This may be especially helpful when repairing a system in a particularly bad state since otherwise the necessary software updates or necessary configuration files may not be fetched for downloading.

13.2.2 Lucid Networking

The Lucid networking functions similarly to the Jaunty networking, but is controlled much more dynamically by the tool `multi_interface_roam`. Rather than removing ip rules manually, `multi_interface_roam` can simply be stopped to put the robot in a nominal non-roaming state. You should only do this if logged in via the service port, since otherwise you will lose your connection.

```
$ sudo /etc/init.d/multi_interface_roam stop
```

One source of debug information related to multi_interface_roam is in the log-file `/var/log/roam/console-output.log`. It shows the 3 available interfaces, and their respective quality metrics, and which one is currently considered active:

```
2011-02-24 21:58:52,147 - INFO - Thu, 24 Feb 2011 21:58:52.147743
2011-02-24 21:58:52,173 - INFO -    bs@lan1   inf             Wired   0.000 100
2011-02-24 21:58:52,173 - INFO -       wan0   inf             Wired 100.000 100 active
2011-02-24 21:58:52,174 - INFO -      ddwrt   inf             Wired  99.998 100 #2
```

13.3 NFS Unionfs

The single largest difference between a normal Ubuntu installation and the PR2 configuration is that c2 mounts nearly its entire filesystem via NFS. The exceptions to this are the directories `/etc`, `/var`, and `/pr2bin` which are mounted via unionfs-fuse. In short, unionfs allows one to specify an additional overlay on top of the underlying filesystem.

On a Jaunty robot, the contents of this overlay can be found in the directory `/slave` on c1. On a Lucid robot, the contents of the overlay can be found in the directory `/unionfs/overlay` on c1. Files added here will show up in the appropriate location on c2. For more information on how this is set up, see the main page for unionfs-fuse and look at the init-script `/etc/unionfs-fuse-nfs-root`.

New software and configuration changes should only be made on c1. Since c2 mounts most of its filesystems read-only, this will usually enforced automatically. If using a non-standard piece of software that attempts to write to the filesystem outside of the home directory, some accommodating changed must be made to either the computer configuration or the software being used.

13.4 autofs

Both of the computers have auto-mount configured for mounting the `/home` partition of the other computer in a computer-independent way using autofs. These auto-mounts are located in the directory `/pr`. To get to the home partition on c1 use the path `/pr/1/` and to get to the home partition on c2 use the path `/pr/2/`. It should rarely be needed to do this explicitly, but it is necessary to make sense of home directory locations.

13.5 Home Directories

The default configuration for user home directories (as given in /etc/passwd) is `/u/username`. Instead of a directory, this location is by default a symlink to `/pr/1/username` (locating the home directory on c1). To place a users home directory at a different location, such as the disk on c2, an admin can simply move (or copy) the home directory and update the symlink accordingly.

13.6 Kernel

Both c1 and c2 run a 2.6 kernel with the RT PREEMPT patch. The kernel is compiled in order to apply the patch and to enable some configuration.

13.7 Storage

Each computer has two hard-drives - one internal 2.5 inch 500Gb Seagate Momentus drive, and a slot for a removable 3.5 inch SATA hard-drive that is exposed on the top of the PR2 base.

By default the only hard-drive used is the internal drive on c1. There are 3 relevant partitions created during installation:

On a Jaunty robot, these are physical partitions:

- c1:`/dev/sda1` - c1:`/` - holds the root filesystem for the OS
- c1:`dev/sda5` - c1:`/home` - stores user home directories (linked from `/u` by way of `/pr/1`)
- c1:`/dev/sda6` - c1:`/hwlog` - stores hardware logs generated by the pr2

On a Lucid robot, these are instead LVM partitions:

- c1:`/dev/vg00/root` - c1:`/` - holds the root filesystem for the OS
- c1:`dev/vg00/home` - c1:`/home` - stores user home directories (linked from `/u` by way of `/pr/1`)
- c1:`/dev/vg00/hwlog` - c1:`/hwlog` - stores hardware logs generated by the pr2

The internal hard-drive on c2 has a single partition, which is generally used as extra user storage:

On Jaunty robot:

- c2:`/dev/sda1` - c2:`/home` - stores user home directories (linked from `/u` by way of `/pr/2`)

On a Lucid robot:

- c2:`/dev/vg00/home` - c2:`/home` - stores user home directories (linked from `/u` by way of `/pr/2`)

Finally, both computers are configured to conveniently make use of the additional removable drives. Any drive loaded into the removable bay should always show up as `/dev/removable`. The mount point: `/removable` is set up to allow users to mount the first partition on this drive.

- `/dev/removable1` - `/removable` - stores temporary data users may want to move off the robot, primarily used for large bag files. This is NOT mounted by default. To use a driver users must explicitly

```
    $ mount /dev/removable
```

to use it, and should

-

```
    $ umount /dev/removable
```

when done. Until the drive is unmounted there is not guarantee that the data has been written to the disk, and if the robot runs out of batteries, data may be lost.

13.7.1 Formatting Drives

To partition/reformat one of the removable drives, or the the internal drive on c2, use the format disk helper utility. Simply pass it the name of the drive that is to be formatted, for example:

```
$ sudo formatdisk /dev/sda
```

or

```
$ sudo formatdisk /dev/removable
```

13.7.2 Drive Maintenance

If a drive is having problems, run a selftest using smartmontools:

```
$ sudo smartctl -t short /dev/sda
```

After a few minutes, check the results:

```
$ sudo smartctl -l selftest /dev/sda
```

```
===START OF READ SMART DATA SECTION===
SMART Self-test log structure revision number 1
Num   Test_Description  Status                   Remaining  LifeTime (hours)
# 1   Short offline     Completed without error  00%        4195
# 2   Short offline     Completed without error  00%        3713
```

In the event that there are reported bad sectors, http://smartmontools.sourceforge.net provides information on freeing the bad blocks.

13.8 Default User Account

The default account on the robot is ”pr2admin”. When the robot is first installed, it does not have a password set, however, this password is set as part of the branding process. When the PR2 is first received from Willow Garage the password will most likely be set to ”willow”.

13.9 Creating User Accounts

The expectation is that each person logging into the robot to develop code will have their own user account. Any robot admin can create a new user using the adduser command (NOT the useradd command), on c1. User accounts are automatically mirrored on c2.

Some examples:

```
$ sudo adduser bob
$ sudo adduser bill -- shell /usr/bin/tcsh --uid 2000
```

Note: it may be helpful to assign users the same UID used on the building network so the UIDs are consistent when mounting shares or moving around the removable drives. Do not create user accounts with UIDs over 30000.

Moving the home directory and creating the symlink is handled by the script:
`/usr/local/sbin/adduser.local`,
which gets run automatically as a part of user-creation. Do not worry about this script at all unless there are changes to the user-creation process.

13.10 User Groups

There are a couple of important groups on the pr2:

- **admin** - Members of this group have full root privileges when using the sudo command
- **rosadmin** - Members of this group have access to change ros-specific configuration settings in /etc/ros
- **apt** - Members of this group can install new software

To add a user to a group, use the usermod command. The most common invocation uses the options -a (append) and -G (group). For example:

```
$ sudo usermod -a -G admin bob
```

13.11 Backing Up and Restoring Users

Before reinstalling the robot operating system back up the user accounts. This can be done with the command: pr2-usermigrate. To save users:

```
pr2admin@c1$ sudo pr2-usermigrate save myrobot.users
```

Move this file off the robot before reinstalling. Then, to restore users:

```
$ sudo pr2-usermigrate load myrobot.users
```

13.12 Clock Synchronization

Consistent time-stamping of data from the two computers is important for interpreting sensor data on a moving system. As a result, keeping the system time on the two computers synced together requires some attention. The system that is used for this is chrony, and the general strategy is to have the two computers tightly coupled to one another, but loosely coupled to an external time source to prevent the robot time from drifting too far from the outside world.

Sometimes the clocks on the computers get into an inconsistent state. This will show up as an error in the pr2dashboard as: ”NTP offset from XXX to YYY; Offset too great.” If the clocks are in an inconsistent state, try restarting chrony on the basestation, c1, and c2, in that order.

```
pr2admin@basestation$ sudo /etc/init.d/chrony restart
pr2admin@c1$ sudo /etc/init.d/chrony restart
pr2admin@c2$ sudo /etc/init.d/chrony restart
```

The following message should show up:

```
Restarting time daemon: chronyd is running and online.
```

If the following message shows:

```
Restarting time daemon: chronyd is running and offline.
```

Restart chrony again.

13.13 Configuring the System ROS Installation

The system ROS installation is managed by a set of configuration files in `/etc/ros`. This configuration determines which distro is used, which ROS processes are launched by default, and contains the calibrated description of the robot URDF. The default configuration files are provided by the package: pr2-core. Additionally, there are packages, pr2-core-boxturtle, and pr2-core-cturtle, which provide specific configurations and dependencies for the different ROS distributions.

In order to change which ROS distro is used by default, simply change the symlink `/etc/ros/distro` to point to a different distribution.

The two important entry point to `/etc/ros` that most users need to know about are `/etc/ros/setup.bash` and `/etc/ros/robot.launch`. Note that each of these is actually just a symlink to the corresponding file in `/etc/ros/distro`. To make use of a specific distribution, use, e.g., `/etc/ros/boxturtle/setup.bash`.

In Jaunty robots, `/etc/ros/setup.bash` works by loading a number of environment variables from files stored in `/etc/ros/distro/env/`. The contents of each of these files will be loaded into the environment variable of the corresponding name.

In Lucid robots, `/etc/ros/setup.bash` instead sources `/etc/ros/distro/setup.sh`. This is where additional environment variables should be set or tweaked if necessary.

`/etc/ros/robot.launch` is responsible for setting up the robot configuration and then launching the minimal set of nodes for robot functionality. The robot.launch files for different distros may need to load different URDFs or analyzer files. It is important that this file should use distrorelative paths rather than depending on the ”distro” symlink. For example, in boxturtle, it will use ”/etc/ros/boxturtle/urdf/robot.xml” rather than ”/etcros/distro/urdf/robot/xml”. This allows the launch file to be used even if the distro symlink points to a different distribution.

Finally, `/etc/ros/sensors/` is a directory containing logical symlinks to sensors. While udev rules have been provided to mount sensors in consistent locations in `/dev/sensors/`, this 2nd layer of indirection allows us to create a symlink such as `/etc/ros/base_hokuyo`, rather than having to modify robot.launch to use something like `/dev/sensors/hokuyo_H0902620`.

The two hokuyos are the most likely devices to be configured incorrectly. If the lasers are switched, swap the symlinks for `/etc/ros/sensors/base_hokuyo` and `/etc/ros/sensors/tilt_hokuyo`. Note that `/etc/ros/sensors` on c1 and c2 are different. To modify the symlinks on c2, modify `/slave/etc/ros/sensors` from the c1 machine.

13.13.1 Switching ROS Distributions Example

Example to switch from ROS boxturtle to ROS cturtle:

```
pr2admin@pr10XX:/etc/ros$ ls -l distro
lrwxrwxrwx 1 root rosadmin      9 2010-11-10 15:27 distro -> boxturtle
pr2admin@pr10XX:/etc/ros$ sudo rm distro
pr2admin@pr10XX:/etc/ros$ sudo ln -s cturtle distro
pr2admin@pr10XX:/etc/ros$ ls -l distro
lrwxrwxrwx 1 root root          7 2010-11-12 10:19 distro -> cturtle
```

13.14 Log Files

The PR2 is configured to consolidate all system logs into `/var/log/messages`.

13.15 Udev

A number of custom udev rules have been provided, to help give sensors more meaningful names. These are all stored in `/etc/udev/rules.d/`. `71-pr1-network-interface.rules` is responsible for renaming the network interfaces and lan0-lan4, wan0, and ecat0 consistently based on mac-address ranges. `73-pr-aliases.rules` makes convenient aliases for the hokuyos, imu, batteries, and joystick. All of these aliases end up in `/dev/sensors/`. Finally `74-pr-disk.rules` is responsible for ensuring that the removable drive always just shows up as `/dev/removable`.

It is important to note that the creation of these udev rules depends on a functioning ROS system. If the ROS configuration from `/etc/ros` is non-functional, udev will not be able to talk to the hokuyos to create appropriate symlinks. In the future there may be more sensors that depend on ROS for this initial configuration as well.

CHAPTER 14

Basestation Setup and Pairing

Before using the PR2 and basestation, it needs to be configured. This configuration can be done by logging into the basestation. The default account is ”pr2admin” with password ”willow”. The following instructions assume that the basestation is plugged into the robot via the service port.

14.1 Requirements

14.1.1 Naming

In order for ROS to work properly, it depends on each computer in the system being able to correctly resolve the hostnames of all other computers in the ROS system. For an in depth explanation, please become familiar with NetworkSetup. In short, 3 hostnames and 3 IP addresses are needed - one for the basestation, and one for each of the two robot computers. Each of these 3 hostnames MUST either be resolved by the building network DNS server, or else they will need to be added to the `/etc/hosts` file of all the relevant computers.

14.1.2 IP Address Configuration

For the robot to be able to find the basestation, it needs to be given a constant IP address. Unless there is a reason not to, this IP address should be assigned to the basestation statically, However, if it is certain that the DHCP server will always assign the same IP, it is ok for this IP to be assigned via DHCP.

The IP address assigned to the basestation must be visible on both the wired and wireless segments of the building network on port 1194. This is the port that VPN uses when the robot makes its connection back to the basestation (it can be changed by appropriately modifying `/etc/openvpn/server.conf` on the basestation, and `/etc/openvpn/basestation.conf` on the robot). Depending on how secure the network is, and whether these segments are separated by a firewall, assistance from the local system administrator may be needed to allow traffic on this port.

Finally, for the PR2 to be able to find the basestation, both the wired and wireless networks must provide a DHCP server. The operation of the system is that the PR2 will acquire an IP address that it is paired to. Review the section on Networking for a more thorough overview.

14.1.3 Examples

In the remainder of the examples, in this section, it is assumed that the basestation has been given the hostname `prbase1`, and the computers will be named prx1 and prx4. It is assumed that the DNS resolves them correctly as follows:

```
prbase1 - 192.168.1.100
prx1 - 192.168.1.101
prx2 - 192.168.1.102
```

14.2 Plugging Things In

Insert Image here Figure: 4.2 ”No Name”

When plugging in the basestation, make sure that WAN0, on the motherboard, is plugged into the building network. LAN0, on the PCI card, is only intended to be plugged into the robot service port. Additionally, it is important to plug the DVI port into the installed graphics card (on the right side of the computer) and not the extra DVI port on the motherboard. There is only one place to plug in the power, and a keyboard/mouse can be plugged into any of the available USB ports.

14.3 Configure the Basestation Network

There are two ways to set up the basestation, using a static IP, or DHCP. Is it strongly recommended that a Static IP is used unless there is a very good reason to use DHCP and it is understood how to configure the DHCP server to obey client-id-based requests.

14.3.1 Configuring IPs

Static IP - To configure the basestation with a static IP edit the file `/etc/network/interfaces` and change wan0 to use a static IP address, for example:

```
iface wan0 inet static
        address 192.168.1.100
        netmask 255.255.255.0
        gateway 192.168.1.1
        post-up robot-forward start
        pre-down robot-forward stop
```

Update `/etc/resolv.conf` to contain the appropriate name server, e.g.,

```
    domain school.edu
    search school.edu
    nameserver 192.168.1.1
```

DHCP - if using DHCP, make sure that the DHCP server respects the client-id specification and is configured to assign a consistent IP address to the client-id of the basestation. **NOTE:** mac address based assignment of IPs on behalf of the robot. The default client-id for the basestation is "basestation". If this needs to be changed to a different client-id, edit the file `/etc/dhcp3/dhclient/conf`, and change the relevant line, for example

```
    send dhcp-client-identifier "prbase1";
```

14.3.2 Configuring Hostname

It is very likely that the hostname of the basestation will need to change. Leaving it as "basestation" is ok, but to make sure the DNS server correctly resolves this name to the IP address that has been assigned to the basestation. To change the hostname edith the files: `/etc/hostname`, to just contain the name of the basestation. For example:

```
    prbase1
```

and edit `/etc/hosts` to resolve the basestation locally. After modifying, `/etc/hosts` it should look something like:

```
127.0.0.1       localhost
127.0.1.1       prbase1.willowgarage.com                prbase1
10.68.255.1     prbase1.willowgarage.com                prbase1

10.68.0.1 c1
10.68.0.2 c2

# The following lines are desirable for Ipv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Finally, set the hostname using the hostname command, for example:

```
$ sudo hostname prbase1
```

14.3.3 Applying Settings

Once the network is reconfigured reboot the basestation to make sure the network comes back up correctly. At this point the user be connected to the local network and be able to ping any machine that the user may want to access the robot from.

14.4 Configure the Robot Wifi

The Linksys wrt610n on the robot is located at the IP address: 10.68.0.5. To configure it, either plug a laptop or a basestation into the robot service port, open a web-browser and a go to: http://10.68.0.5. Click on the "Wireless" tab. The default login should be "root" and "willow", although after a factory reset the router may end up with login, "root" and "admin". Set either "Wireless Physical Interface wl0" (for 2.5 Ghz) or "Wireless Physical Interface wl1" (for 5 Ghz) to client mode and enter the SSID for the network. To change other settings, make sure that under "Setup", the WAN connection type is "Automatic Configuration - DHCP", the local IP Address is 10.68.0.5, and the DHCP Server is Disabled. When done, make sure to select "Apply Settings".

To test the setup of the wireless, you can temporarily add a specific route on the basestation which will route a particular IP address through the wifi router, and then try to access that ip address. Once the routing entry is added and the IP address is reached, the wifi router is connected properly. For example, to route requests to the Willow Garage website through the wifi-router explicitly:

```
$ sudo ip route add 157.22.19.22 via 10.68.0.5 dev lan0
```

Then point the web-browser to: http://157.22.19.22. This should end at Willow Garage's home page via the wireless router. If the building-infrastructure wireless access point requires accepting an EULA, this method will typically give an opportunity to accept it.

Once that is working, remember to disable the route so that it works if the basestation is not plugged into the service port:

```
$ sudo ip route del 157.22.19.22 via 10.68.0.5 dev lan0
```

14.5 Initialize the Basestation VPN Server

One of the primary functions of the basestation is to host a VPN server that the robot can connect to. On Jaunty robots this configuration is in `/etc/openvpn`. On Lucid robots this configuration is in `/etc/wg-openvpn`.

The server needs an appropriate key. To generate they key for the VPN server on Jaunty, run:

```
$ sudo /etc/openvpn/gen_server_key
```

on Lucid, run:

```
$ sudo /etc/wg-openvpn/gen_server_key
```

It will prompt for some information. It is good practice to get the information correct, but will not actually impact the performance of the robot.

14.6 Pairing with the Robot

Robot pairing is performed with the robot-brand command on the basestation. It takes 5 arguments:

```
$ sudo robot-brand <ROBOT_NAME> <C1_NAME> <C2_NAME> <VPN_SUBNET> [<BASESTATION_IP>]
```

- ROBOT_NAME - This is the name the PR2 will be given.
- C1_NAME - This is the intended hostname of c1. The DNS server needs to resolve this to the IP address that is intended to be used for the PR2.
- C2_NAME - This is the intended hostname of c2. The DNS server needs to resolve this to the IP address that is intended to be used for the PR2.
- VPN_SUBNET - Unless multiple PR2's are being used 10.68.1.0. The 3rd field of the IP, however, can be any value other than 0 and 255, which are reserved for the robot local network and the basestation server respectively. No two robots on the same basestation should be assigned the same VPN subnet.
- (OPTIONAL) Basestation_IP - By default this is taken from the IP assigned to wan0. It is worth checking that it is correct using the ifconfig command:

```
$ ifconfig wan0
wan0            link encap:Ethernet  Hwaddr 00:30:1b:48:c8:f8
                inet addr:192.168.1.100 Bcast:192.168.1.255  Mask:255.255.255
                inet6 addr: fe80::230:1bff:fe48:c8f8/64 Scope:Link
                UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                RX packets:2702 errors:0 dropped:0 overruns:0 frame:0
                TX packets:2709 errors:0 dropped:0 overruns:0 carrier:0
                collisions:0 txqueuelen:1000
                RX bytes:129699 (129.6 KB) TX bytes:218528 (218.5 KB)
                Interrupt:17
```

In this case the address is 192.168.1.100

To actually brand the PR2, first plug the basestation into the service port of the PR2, and then use the robot-brand command. For example, if the Pr2 would be named

”prx”, c1 would be named ”prx1”, c2 would be named ”prx2”, and if wanting to use the default 10.68.1.0 192.168.1.100, run:

```
$ sudo robot-brand prx prx1 prx2 10.68.1.0
```

During the branding process, it may prompt for the password of ”root@c1”. By default this will be ”willow”. Additionally, a prompt may occur to choose a password for the pr2admin account. When branding a Lucid robot, you will additionally be asked to confirm multiple steps of the branding process. Unless you have reason not to, you should select ”y” at each prompt. When the script finishes a user should be able to connect to the robot over the VPN. Check that it works and then exit.

```
$ ssh pr2admin@10.68.1.1
$ exit
```

14.7 Forwarding IPs to the robot

To make contacting the robot from the building network more convenient, the basestation can be configured to forward IP addresses (assigned statically or via DHCP) to the robot. This is achieved through a combination of ip-tables rules and additional ip addresses claimed by the basestation.

First the basestation should claim a couple extra IP addresses. These are the IPs that will be used to contact the robot from the local network. Eventually these IP addresses should show up in ifconfig, for example:

```
wan0:prx1 Link encap:Ethernet Hwaddr 00:30:1b:48:c8:f8
                inet addr:192.168.1.101 Bcast:0.0.0.0 Mask:255.255.255.255
                UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                Interrupt:17

wan0:prx2 Link encap:Ethernet Hwaddr 00:30:1b:48:c8:f8
                inet addr:192.168.1.102 Bcast:0.0.0.0 Mask:255.255.255.255
                UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                Interrupt:17
```

Additionally, ip-tables should forward each of these IP addresses to the respective robot computer in the VPN. In the IP tables (viewable with iptables-save):

```
-A PREROUTING -d 192.168.1.101/32 -j DNAT --to-destination 10.68.1.1
-A PREROUTING -d 192.168.1.102/32 -j DNAT --to-destination 10.68.1.2
-A POSTROUTING -s 10.68.1.1/32 -j SNAT --to-source 192.168.1.101
-A POSTROUTING -s 10.68.1.2/32 -j SNAT --to-source 192.168.1.2
-A OUTPUT -d 192.168.1.101/32 -j DNAT --to-destination 10.68.1.1
-A OUTPUT -d 192.168.1.2/32 -j DNAT --to-destination 10.68.1.2
```

Setting this up by hand is tedious and time-consuming. Instead a utility called robot-forward is provided which takes care of setting this up automatically. To set this up, edit the file: `/etc/robot-forward.conf` and add the lines:

```
<C1_NAME> <VPN_SUBNET>.1 <ROBOT_IP1>
<C2_NAME> <VPN_SUBNET>.2 <ROBOT_IP2>
```

For example, in this case:

```
prx1 10.68.1.1 192.168.1.101
prx2 10.68.1.2 192.168.1.102
```

Note that for this to work properly, the PR2 computers should have been named prx1 and prx2, respectively, and furthermore the DNS server should resolve ”prx1” to 192.168.1.101, and ”prx2” to 192.168.1.102.

Much like the basestation, these IPs can be assigned via DHCP:

```
prx1 10.68.1.1 dhcp
prx2 10.68.1.2 dhcp
```

In which case the basestation will use the client-id’s ”prx1” and ”prx2” to acquire 2 IP addresses from the DHCP server. These IPs should be assigned consistently given the client-id’s, and should match the IPs that the DNS server returns for the particular hostnames. Robot forwarding is enabled whenever the network starts, but to manually start forwarding, run:

```
$ sudo robot-forward start
```

The user should now be able to contact the robot at the given IP address, and, if they DNS server is set up properly, using its hostname. For example:

```
$ ssh par2admin@192.168.1.101
```

and

```
$ sudo robot-forward stop
```

At this point the basestation should be configured to work on the network.

CHAPTER 15

Reinstalling the Basestation

With a basestation USB installer, the basestation can be reinstalled to revert it to a completely fresh state. Creating a new USB installer is outside the scope of this manual, but instruction for downloading and creating one can be found on the PR2 Support website under Download.

15.1 Booting from the USB Installer

To boot off the USB installer:

1. Insert the USB stick into one of the USB ports in the basestation.
2. Reboot the basestation.
3. Press the DEL or Delete key to enter setup.
4. Choose ”Advanced BIOS Features”.
5. Choose ”Hard Disk Boot Priority”.
6. Select the drive listed as ”USB-HDD0” and press ”Page Up” to make it number 1.
7. Press ”F10” to save and exit the setup.
8. The computer should boot off of the USB stick.
9. If the installer complains about not being able to find the ISO, choose ”Scan Hard Drive fro ISO...” and then press ”Enter” to continue.
10. Once the installer is actually progressing, it will take about half an hour to complete.
11. When the installer completes, press ”Enter” to shut down and remove the USB stick.

12. When the basestation boots a fresh image should be acquired the section on Basestation Setup and Pairing to configure the basestation.

15.2 Booting from the DVD Installer

To boot off the DVD installer:

1. Insert the DVD into the drive on the basestation.
2. Reboot the basestation.
3. Press the DEL or Delete key to enter setup.
4. Choose ”Advanced BIOS Features”.
5. Choose ”Hard Disk Boot Priority”.
6. Select the ”CDROM” and press ”Page Up” to make it number 1.
7. Press ”F10” to save and exit the setup.
8. The computer should boot off of the DVD.
9. If the installer complains about not being able to find the ISO, choose ”Scan Hard Drive for ISO...” and then press ”Enter” to continue.
10. Once the installer is actually progressing, it will take about half an hour to complete.
11. When the installer completes, press ”Enter” to shut down and remove the DVD.
12. When the basestation boots a fresh image should be acquired - consult the section on Basestation Setup and Pairing to configure the basestation.

CHAPTER 16

Reinstalling the Robot

To reinstall the robot, first back up anything needed off of c1. Next, shutdown the computer using pr2-shutdown –netboot. Make sure to then plug the service port directly into the basestation. Run the robot installer:

```
$ sudo robot-install
```

1. Boot c1 by pushing the power button with a paperclip.

2. After N minutes, the net-console on the basestation will show the install dialog. Press ”Enter” to start the installer and wait for it to complete.

3. Once the installer has completed and the PR2 is shut down, boot it up. The PR2 must be paired with the basestation before it can be used.

16.1 Forcing a Netboot

If the robot BIOS is not configured to netboot, hit F12 on the keyboard. This can be done in one of two way. The easier way is to connect up to the KVM. (See the section on how to connect up to the KVM).

Alternatively, plug a keyboard into the PR2. This can be done using the aux usb port under the back cover. To remove this cover, follow the instructions found in the PR2 Support website.

When the PR2 is booting 2 beeps will sound followed by a short pause, and then another 2 beeps. After the 4th beep, hit F12 on the keyboard (either physically or through the KVM) to force c1 to neboot.

CHAPTER 17

Software Maintenance

When the robots ship, the most recent version of ROS and the PR2 software stacks is installed in /opt/ros. By default, new users will have a ROS installation which references the pre-installed code. Users who wish to modify or replace a part of the system are recommended to install a development version of the stacks or packages which they wish to modify, but to continue to use the base installation for most of the robot functionality.

Debian packages exist for all of the custom software and configurations on the robot. The package which pulls in all the other requirements is called pr2-environment. pr2-environment depends on pr2-core, which in turn depends on ros-boxturtle-pr2, the meta-package which contains all of the ROS software for the PR2. To upgrade the software on the PR2 simply use apt:

```
$ sudo apt-get update
$ sudo apt-get dist-upgrade
```

Which will update all packages on the system, pulling in security fixes for the pr2 configuration as well as for the appropriate ROS-release.

17.1 Fixing the Nvidia Module

Occasionally upgrading the basestation will cause a new kernel to be installed. When this happens, the nvidia module which are built for the kernel will no longer work. If this happens, once a reboot is done, X will fail to start. Choose the option to exit X. The script to re-build and install the modules is located in `/usr/src/NVIDIA-Linux-x86_64-195.36.15-pkg2.run`. Simply run it:

```
$ sudo /usr/src/NVIDIA-Linux-x86_64-195.36.15-pkg2.run
```

CHAPTER 18

Running the PR2

Running the PR2 requires a basic understanding of ROS, the BSD-licensed Robot Operating System. A ROS system consists of multiple processes running on multiple computers. If ROS is not familiar, it is recommended to follow some of the beginner tutorials on ros.org. Familiarity with ROS tools will make using the robot much easier. In particular, the PR2 user should understand what a launch file is and how to run it, and how to run ROS with nodes on multiple computers. This chapter will walk through the steps of starting up and running a PR2, using ROS.

18.1 Out of the Box

Please see the video Unpacking the PR2. This chapter assumes that the PR2 is already set up for use in the lab.

18.2 Batteries and Power

Before running the robot, make sure it has power. Follow the instructions in this chapter to start up the PR2 while it is plugged into the power outlet, which will keep the batteries charged. The battery life of the PR2 is approximately two hours, so it is a good idea to keep the PR2 plugged in when not in use.

18.3 Run-Stop

Before running the PR2 always have the wireless run-stop nearby in case the motors need to be powered off for any reason. The PR2 has two run-stop buttons: the red push-button on the middle of the PR2's back, and the yellow wireless run-stop transmitter. If either one is in the stop position, the motors are disabled so the robot cannot move. In an emergency, press the stop button on the wireless transmitter, or hit the push-button in the back. Putting either run-stop button in the stop position does not damage the PR2 or turn off the computers, it simply stops the motors.

The wireless run-stop is also powered by batteries, which can run down. Therefore, it is a good idea to turn the wireless run-stop off (by pressing the "STOP" button) when not in use. If the wireless run-stop runs out of battery charge, then the battery light (the light next to the battery symbol in the lower half of the wireless run-stop face) will flash. To change the battery, a slotted screwdriver is needed to open the wireless run-stop case.

18.4 Getting an Account

Before using the PR2, an account on the robot computers is needed. Ask the robot administrator to create an account for PR2 users, using the instructions in Chapter 13.

18.5 Turning on PR2

If the robot is already running skip this step, but the wireless run-stop should still be in the stop position.

To turn the PR2 on:

1. Verify that the wireless run-stop is in the stop position. If it is not yet in the stop position, they press the wireless run-stop's red "STOP" button; the light s on the wireless run-stop should turn off.

2. Switch the red run-stop button on the back panel of the robot to the "on" position; turn it clockwise and it will pop out.

3. If the DC breaker switch at the back panel of the robot is switched off, then flip the DC breaker switch (Figure 18.1) to the "on" position.

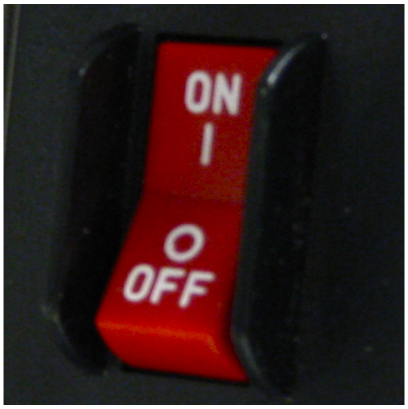

Figure 18.1: The PR2 DC breaker switch in the ON position.

The red lights on the computers should turn on, the fans should spin up, and several rising-tone beeps from the computers should be heard as they boot. The process of booting the computers will take about five minutes. The computers are on when a red light illuminates on each of the computers in the robot base.

18.6 Logging In

Once the PR2 is on, ssh into the main computer using the account created by the administrator, using the steps below. In the following instructions, prx1 will be used to refer to the first computer on robot x and username to refer to the account username. Ask the robot administrator what name should be used for the PR2 being set up.

To log into the robot, open a terminal window and type

```
$ ssh <username>@pr<x>1
```

Access to all the ROS tools should be available. For instance, to see what packages are available, type

```
$ rospack list-names
```

and a list of the ROS packages currently in the path should be visible.

18.7 Checking for Other Users

Always check to see if anyone else is using the PR2, if logging into a robot that was already running.

```
$ robot users
```

This will show who is currently running robot. If nothing shows up, then no one else is logged into the PR2. If other people are logged into the robot, find out if their work may be interrupted. Ask if it is ok to kill their processes so that new work can be done. If the people running processes on the robot cannot be found, ask the robot administrator for guidance on what the policy is in the lab. If it is fine to kill other processes running on the robot, type

```
$ robot stop
```

and all the processes that are being run on the robot will be killed.

18.8 Starting the PR2 Software

To start the PR2, use the robot command again to first claim the robot by leaving a message that others can see when logging into the robot.

```
$ robot claim -m "my message"
```

Then use the robot command to launch the basic functionality of the PR2. This includes drivers for the sensors, motors, speakers, projector, power system, and joystick as well as the default set of realtime controllers, processing and logging of diagnostics information, and monitors for various types of problems the PR2 could experience. **NOTE:** When using the robot command to start the PR2 the user's current ROS_PACKAGE_PATH will be used for the default path.

```
$ robot start
```

18.9 PR2 Dashboard

When running the PR2, the pr2_dashboard should always be up on the screen. This is how the robot software will announce if something is going wrong, and is also how the motors and power turn on and off.

On a computer with a built ROS installation (e.g., the basestation desktop computer that ships with the PR2), not on the PR2 itself, set the ROS_MASTER_URI to point at the master running on the PR2 and launch the dashboard by typing:

```
$ export ROS_MASTER_URI=http://pr<x>1:11311
$ rosrun pr2_dashboard pr2_dashboard
```

The pr2_dashboard control panel (a graphical user interface) should appear and provide information about the state of the PR2. It is ok if not all of the icons are green. In fact, the run-stops should be ”warning” that the wireless run-stop is not on.

Take a moment to review the state of the PR2. A sense can be obtained for the health of the PR2 by looking at the diagnostics information; click on the wrench on the far left. Since the run-stop is in the stop position, the motors will be giving a warning because the robot was just turned on and the encoders on the joints have not been calibrated yet.

If warnings or errors are seen in any other sections, read the error messages and try to figure out what the problem is. As the PR2 administrator for help if there are errors that are not understood.

For more information on using and debugging with the pr2_dashboard see the beginner PR2 tutorials on pr2support.willowgarage.com.

18.10 Calibrating the Joints

If no problems are discovered in the diagnostics, and the continuation of the PR2 bring-up is desired, first check the physical area around the robot to ensure that the PR2 will not hit anything around it when it calibrates. The PR2's arms will be fully extended and the spine will move up and down during calibration. Once the physical area is cleared, press the green ”START” button on the wireless run-stop. This should change the state of the run-stop indicator on the dashboard to a green circle, and will allow the robot to turn power on for the motors.

If the PR2 circuit breakers are not yet enabled, the next step is to enable the circuit breakers (marked L, B, R). When the circuit-breaker icon is clicked, select ”enable all” from the menu to turn on all three breakers.

If the PR2 motors are not yet reset, then click on the picture of a motor, and select ”reset motors”.

When the wireless run-stop is started all the PR2 circuit breakers are enabled, and the PR2 motors are reset, the PR2 will move its joints to find the absolute reference positions of each joint so that it can calibrate that mechanism. When calibration is finished, all the icons in the dashboard should be reading OK.

If this calibration step fails, then use the pr2_dashboard to diagnose the problems with the PR2. See the PR2 administrator if assistance is needed in understanding and addressing the issues.

18.11 Tucking PR2 Arms

Before driving the PR2 around, it is best to tuck the PR2's arms, If the PR2's arms are not tucked in before driving, then the arms are likely to swing around while the PR2 is moving.

To run the tuckarm package, type into the PR2's computer (prx1):

```
$ rosrun pr2_tuckarm tuckarm.py b
```

This will tuck both PR2 arms. The feedback from tuckarm should display something like this:

```
[INFO] 1264099125.710039: Waiting for controller manager to start
[INFO]  1264099125.718283: Tucking both left and right arm
```

For more information about how to drive the robot around, see the pr2_tuckarm package at ros.org.

18.12 Driving PR2 with a Joystick

To move the PR2, drive it around with the joystick. Tod o this, open a new terminal window with a new ssh connection to the robot and run the teleop_joystick launch file.

```
$ ssh <unsername>@pr<x>1
$ roslaunch pr2_teleop teleop_joystick.launch
```

Be sure to check to see if the PR2 is plugged in (e.g., ethernet cables or power cables). If PR2 is plugged in, be very cautious to avoid pulling the cables out improperly.

Once the teleop_joystick mode is running, press the pairing button in the middle of the joystick to pair it with the PR2. Be sure to unplug the PR2 from wall outlets if the PR2 is going to be driven around.

For more information on the buttons and functionality provided by the joystick see the pr2_teleop package.

18.13 Visualizing Sensor Data

To see the sensor data from the PR2, use rviz on the off-board computer (e.g., base station). Open a new terminal window and configure the ROS_MASTER_URI to point at the PR2, run

```
$ export ROS_MASTER_URI=http://pr<x>1:11311
$ rosrun rviz rviz
```

and rviz should launch with a visualization of the PR2.

For more information about how to view different types of data coming from the PR2, see the rviz documentation at ros.org.

18.14 Putting PR2 Away

To properly put away the PR2, drive the PR2 to the appropriate location in the lab, plug in the PR2 to recharge the batteries, press STOP on the wireless run-stop, stop the process on the PR2, and log off of the PR2.

- Put away the PR2. Drive the PR2 to the appropriate location in the lab and plug it in to recharge.
- Stop all the processes running on the PR2. To stop all the processes running on the PR2 and log off of the PR2, but do not turn off the PR2 computers, then open a new terminal and type:

```
$ ssh <username>@pr<x>1
$ robot stop
```

- Exit from the PR2's computer terminal, type:

```
$ exit
```

If someone was waiting to use the PR2, be sure to let them know that the PR2 is no longer in use.

As an alternative to simply stopping the processes running on the PR2, turn off the PR2 computers. To do this, open a new terminal and type:

```
$ ssh <username>@pr<x>1
$ sudo pr2-shutdown
```

If the PR2 is turned off, four sets of descending beeps from the PR2's computers will be heard. Then the red lights on each of the computers will turn off. If this does not happen, then the PR2's computers might not be completely shut down properly.

CHAPTER 19

PR2 SE