CoroControl for Linux

May 7

2013

This document describes the steps that have to be followed in order to remotely control a CoroBot from a computer.

User's Guide

CoroControl for Linux

Version 17

Table of Contents

1. INTRODUCTION 3
2. REMOTE CONTROL 3
2.1. STARTING UBUNTU 3
2.1.1. STARTING FROM THE DVD 3
2.1.2. UBUNTU INSTALLATION 5
2.1.2.1.INSTALLING FROM THE DVD 5
2.2. COMPUTER CONFIGURATION 7
2.3. ROBOT CONFIGURATION 10
2.3.1. ENVIRONMENT SETUP 10
2.4. ROBOT CONTROLS 10
2.4.1. DIRECT ROBOT CONTROLS 11
2.4.2. REMOTE CONTROL CONFIGURATION 12
2.4.2.1.CONFIGURATION STEPS: 12

Table of Figures

FIGURE 1 - COROBOT 2
FIGURE 2 - UBUNTU LAUNCH SCREEN 4
FIGURE 3 - UBUNTU DESKTOP 5
FIGURE 4 - INSTALL SCREEN 5
FIGURE 5 - PREPARING TO INSTALL UBUNTU 6
FIGURE 6 - SELECT INSTALLATION TYPE 6
FIGURE 7 - PARTITION MODIFICATION 7
FIGURE 8 - UBUNTU NEW INSTANCE 7
FIGURE 9 - DESKTOP WITH TERMINAL ICON 8
FIGURE 10 - TERMINAL 8
FIGURE 11 - COROBOT REMOTE CONTROL INTERFACE LAUNCH 9
FIGURE 12 - COROBOT-REMOTE CONTROL INTERFACE 10
FIGURE 13 - TEXT EDITOR 12

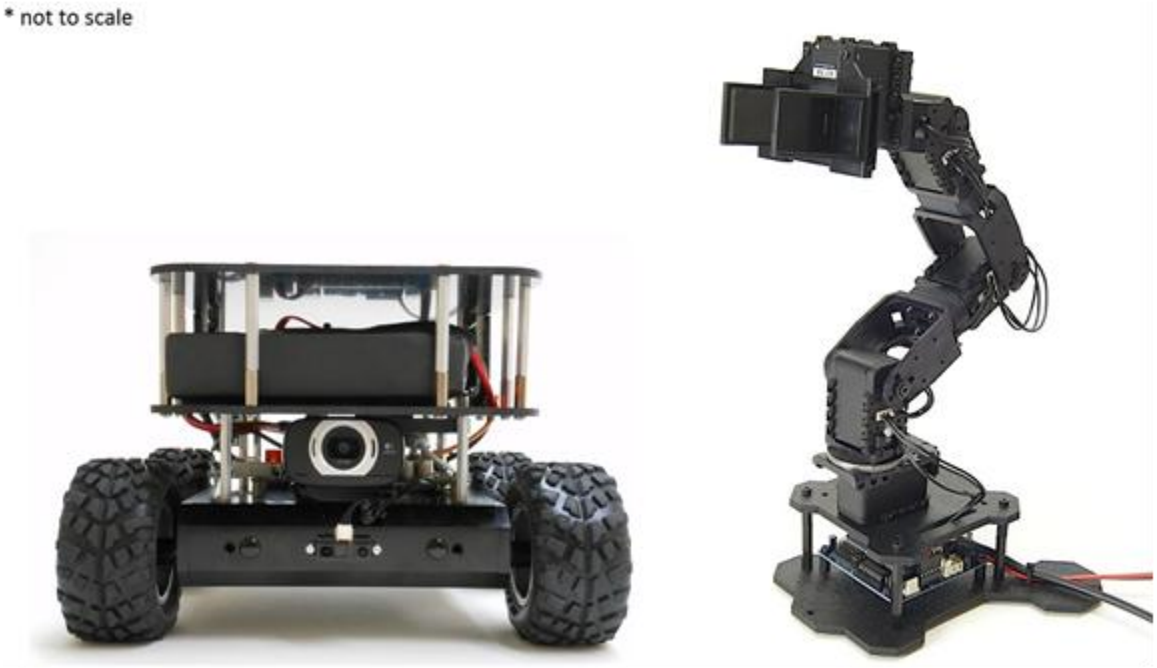


Figure 1 - CoroBot

This document is for informational purposes only.

1. Introduction

This document describes the steps that have to be followed in order to remotely control a CoroBot from a computer. In this document we will refer to the CoroBot you want to control as “robot” and the remote computer that is used to control the robot as “computer”. This distinction has to be done because the robot is also a computer.

2. Remote control

2.1. Starting Ubuntu

First you need to start Ubuntu on the computer. In order to do that you can easy install Ubuntu or just start it from the custom live DVD, downloadable at https://dl.dropbox.com/u/3246099/livecd.iso. This document explains both methods.

For this procedure you might need to put the iso file onto a bootable flash drive. The method is differs with the operating system you are working on so please follow one of the links to create the bootable flash drive:
http://www.ubuntu.com/download/desktop/create-a-usb-stick-on-ubuntu
http://www.ubuntu.com/download/desktop/create-a-usb-stick-on-mac-osx
http://www.ubuntu.com/download/desktop/create-a-usb-stick-on-windows

2.1.1. Starting from the DVD / Flash Drive

The advantage of this method is that it is moderately fast to execute and get ready to control the robot, but all the steps presented here have to be done every time you want to control the robot.

Follow these steps to start from the CoroWare Ubuntu DVD:
1) Insert the DVD in the drive or the bootable flash drive in a USB port.
2) Start the computer and make sure to be able to start from the DVD or USB.
3) Select “Try Ubuntu without installing”.

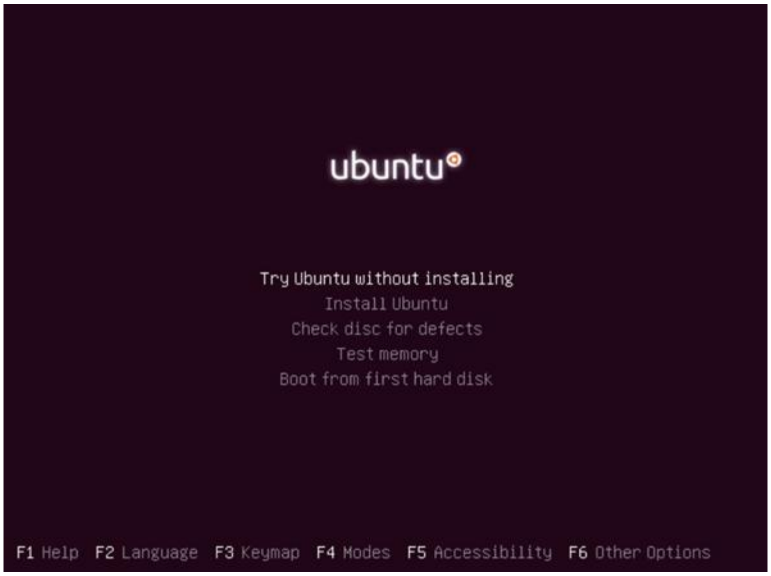


Figure 2 - Ubuntu launch screen

4) You now are on Ubuntu Desktop and your screen should look like the following image

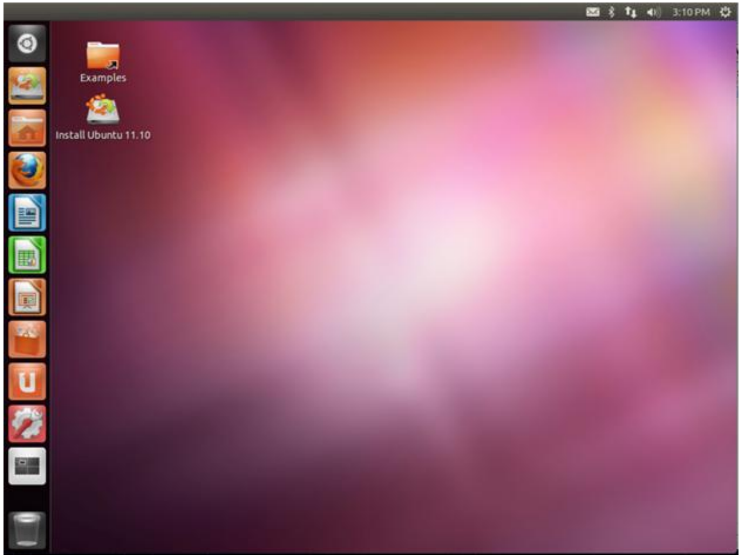


Figure 3 - Ubuntu Desktop

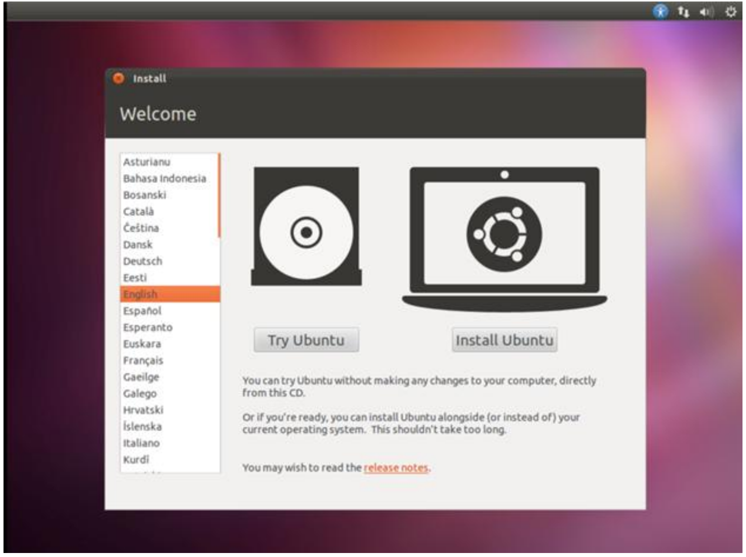
2.1.2. Ubuntu Installation

This method has the disadvantage to take a long time the first time it is executed but is very fast every other time.

2.1.2.1. Installing from the DVD

Follow these steps to start from the CoroWare Ubuntu DVD:

1) Insert the DVD in the drive or the bootable flash drive in a USB port
2) Start the computer and make sure to be able to start from the DVD or USB
3) Select install Ubuntu
4) At the following screen, choose your preferred language and select “Install Ubuntu”



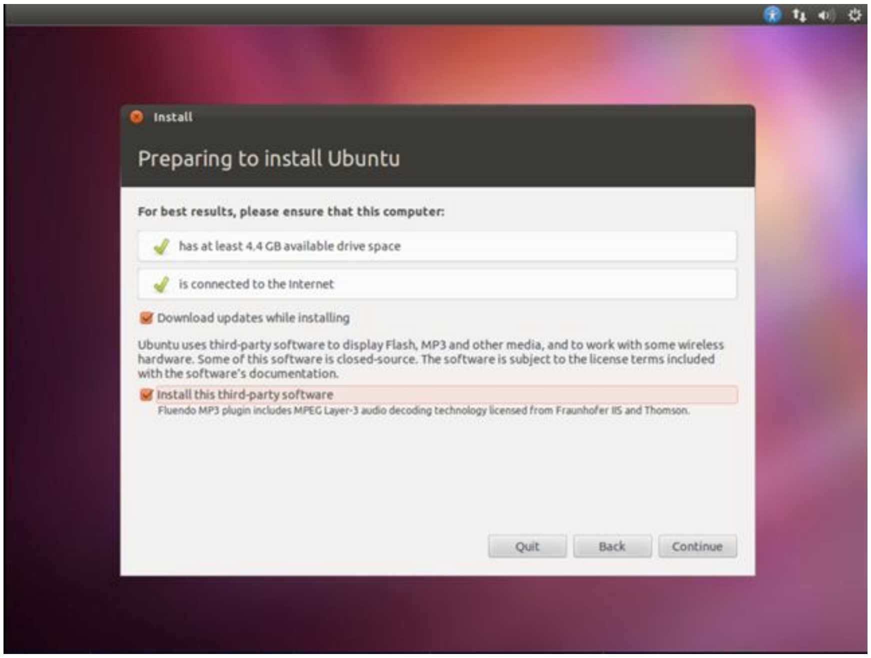
You can try Ubuntu without making any changes to your computer, directly from this CD.

Or if you're ready, you can install Ubuntu alongside (or instead of) your current operating system. This shouldn't take too long.

You may wish to read the release notes.

Figure 4 - Install Screen

5) Then, check both boxes and click on "Continue"



Install

Preparing to install Ubuntu

For best results, please ensure that this computer:

✓ has at least 4.4 GB available drive space

✓ is connected to the Internet

☑ Download updates while installing

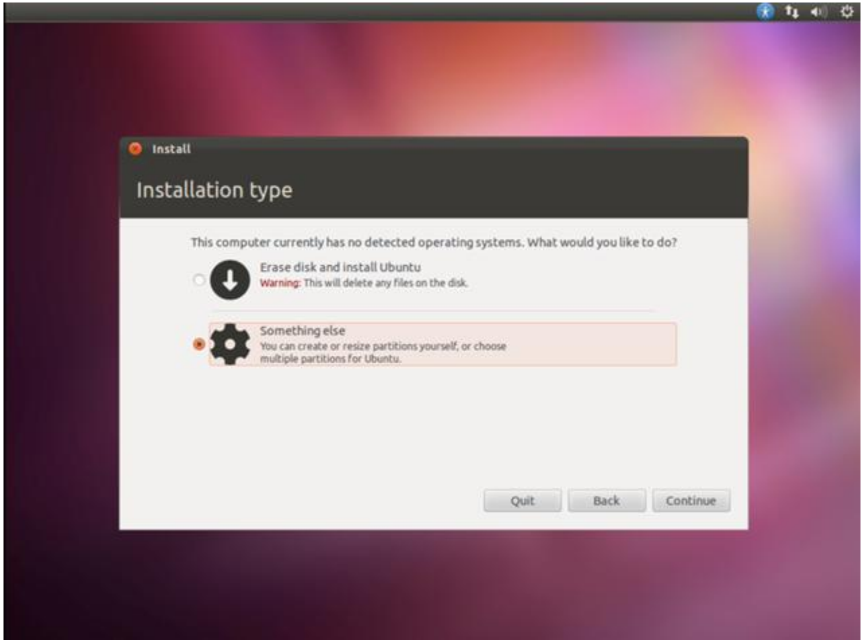
Ubuntu uses third-party software to display Flash, MP3 and other media, and to work with some wireless hardware. Some of this software is closed-source. The software is subject to the license terms included with the software's documentation.

☑ Install this third-party software

Fluendo MP3 plugin includes MPEG Layer-3 audio decoding technology licensed from Fraunhofer IIS and Thomson.

Figure 5 - Preparing to install Ubuntu

6) Select "Something Else" and click on "Continue"



Install

Installation type

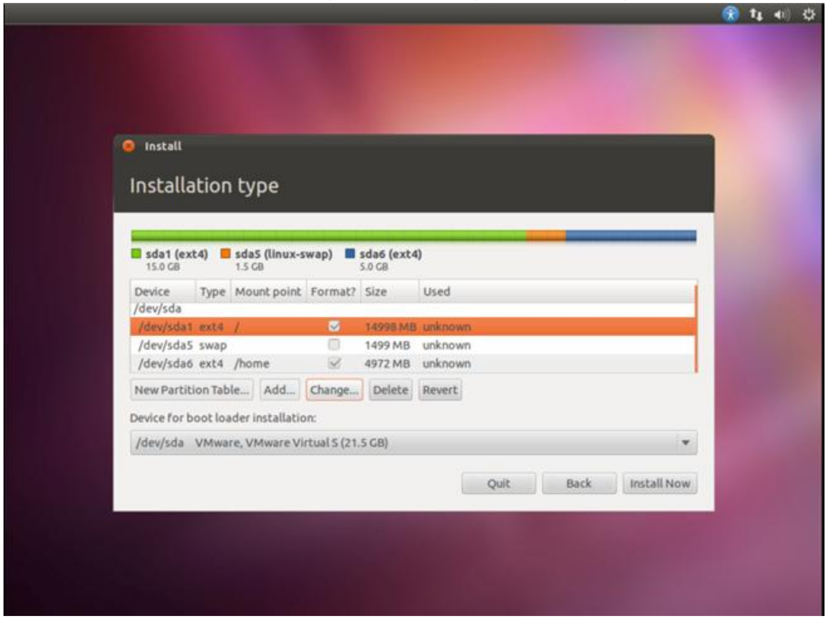
This computer currently has no detected operating systems. What would you like to do?

○ Erase disk and install Ubuntu
Warning: This will delete any files on the disk.

● Something else
You can create or resize partitions yourself, or choose multiple partitions for Ubuntu.

Figure 6 - Select installation type

7) Modify and create the necessary partitions, select the journaling file system type, select the mount points and click on install now. For more information, go to: https://help.ubuntu.com/10.04/switching/C/installing-partitioning.html



Install

Installation type

🟩 sda1 (ext4) 15.0 GB 🟧 sda5 (linux-swap) 1.5 GB 🟦 sda6 (ext4) 5.0 GB

Device	Type	Mount point	Format?	Size	Used
/dev/sda					
/dev/sda1	ext4	/	☑	14998 MB	unknown
/dev/sda5	swap		□	1499 MB	unknown
/dev/sda6	ext4	/home	☑	4972 MB	unknown

Device for boot loader installation:

/dev/sda VMware, VMware Virtual S (21.5 GB)

Figure 7 - Partition modification

8) Follow the instructions appearing on the screen and wait until the installation finishes before clicking on restart.

2.2. Computer configuration

1) Click on the Ubuntu icon on the top left corner, a new screen appears that look like this:

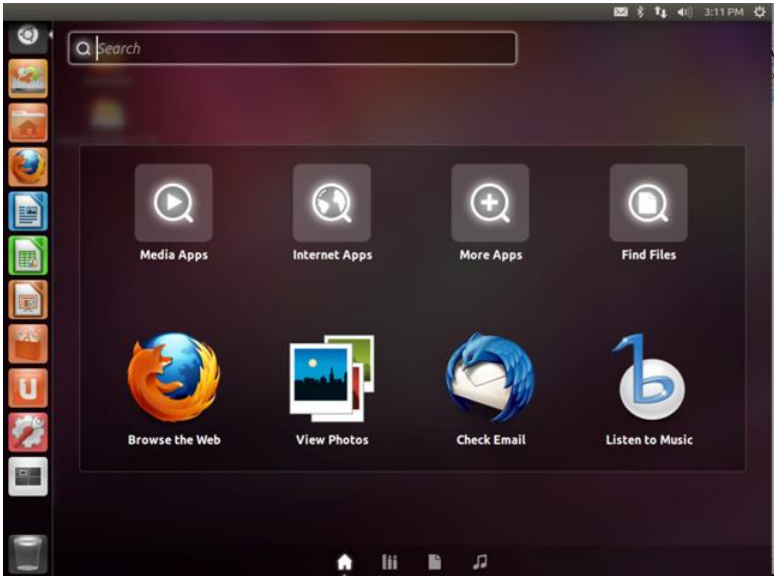


Figure 8 - Ubuntu new instance

2) Now you can write “terminal” in the research area. The terminal icon should appear on the screen, as shown below:

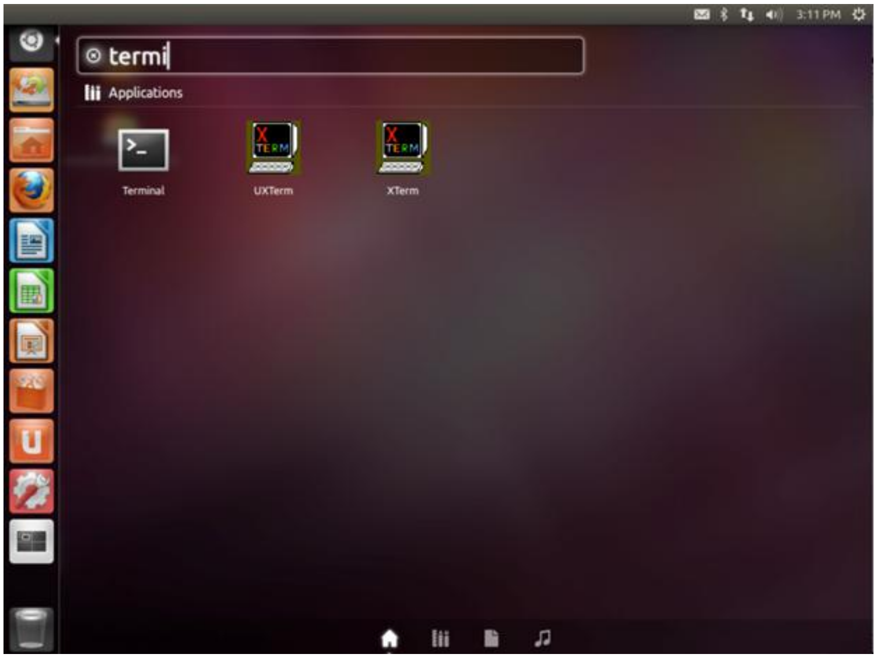
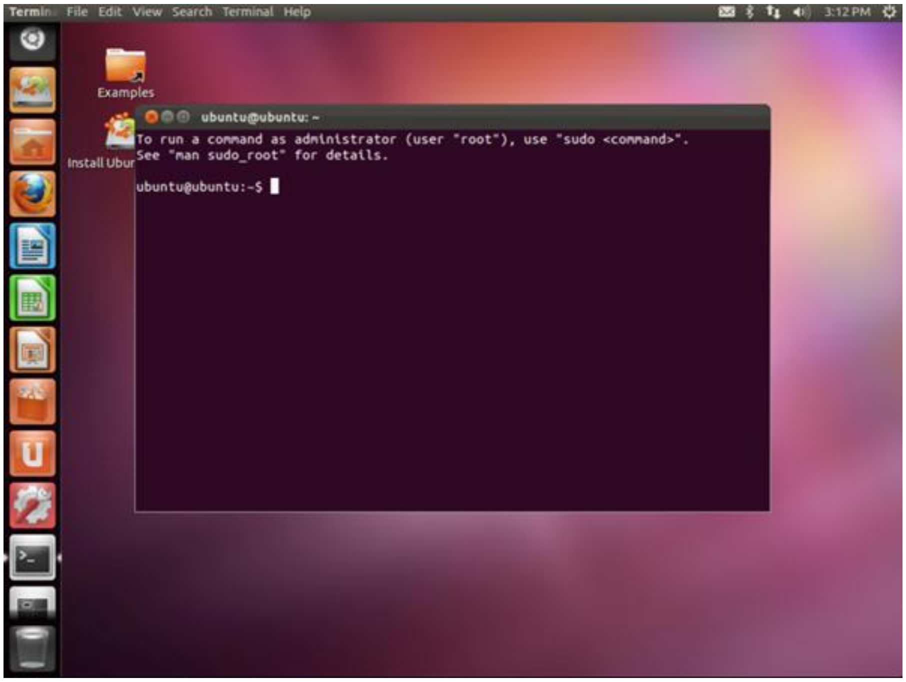


Figure 9 - Desktop with Terminal icon

3) Left click on the terminal icon to launch it.



```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ubuntu:~$
```

Figure 10 - Terminal

4) Now we need to configure ROS. A couple of commands need to be written in this terminal, each time you write a command press enter to execute it before writing the next one. For more clarity each command is separated by a line in the list given bellow:

```
echo “source /opt/ros/fuerte/setup.bash” >> ~/.bashrc

source ~/.bashrc

echo “export ROS_PACKAGE_PATH=$ROS_PACKAGE_PATH:~/ros_packages/” >> ~/.bashrc

source ~/.bashrc

svn checkout svn://svn.code.sf.net/p/roscorobot/code/trunk/Electric ros_packages

rosmake –pre-clean Corobot

rosrun corobot_teleop corobot_teleop
```

5) If all the commands were executed properly, a window should appear and your screen should look like this

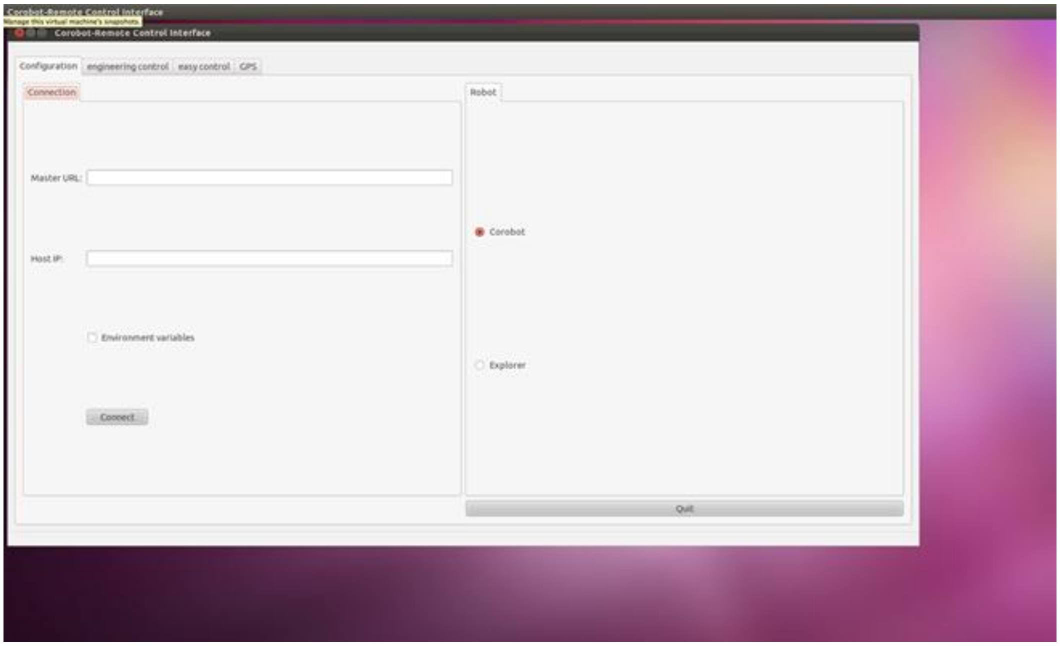


Figure 11 - CoroBot remote control interface launch

2.3. Robot configuration

2.3.1. Environment Setup

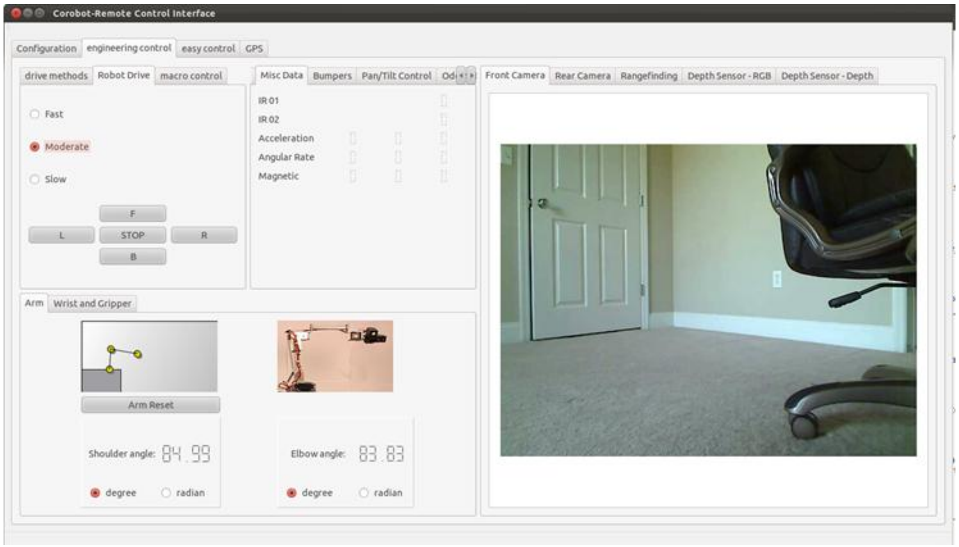
1) Open a terminal and enter the command “gedit ~/.bashrc”.
2) At the end of the file, if not present already, add the following lines:
    1) export ROS_MASTER_URI=http://localhost:11311
    2) export ROS_IP=xxx.xxx.xxx.xxx (Robot IP)

3) Save the file and close the terminal
4) Open a new terminal and write the command “roslaunch corobot_teleop corobot_ros_gui.launch”
5) The same window that appeared on the computer should appear. This window can be closed.

Note: *If you wish to control the robot manually, all you have to do is click on the connect button in the remote control software that appeared in this last step and skip every step related to the remote computer.*

2.4. Robot Controls

Now to control the robot you need to go back to the opened window on the computer and fill in the master url and host ip section. The master url is in the form of http://xxx.xxx.xxx.xxx:11311 (the x representing the ip of the robot), the host ip is the ip of the computer. Just click connect and go to the other tabs to control the robots. The window should look like the following image.



○ Fast
● Moderate
○ Slow

Shoulder angle: 84.99
● degree ○ radian

Elbow angle: 83.83
● degree ○ radian

Figure 12 - CoroBot-Remote Control Interface

Note: You can skip this step by opening the file ~/.bashrc using the gedit command and add the following lines at the end of the file:

- Export ROS_MASTER_URI=http://xxx.xxx.xxx.xxx:11311 (Robot IP)
- Export ROS_IP=xxx.xxx.xxx.xxx (Computer IP)

REMOTE CONTROL INTERFACE CONTROLS:

The software contains several tabs, each with a different purpose:

1) Configuration Tab – User can set up the connection with the robot
2) Engineering Control Tab – user can control the robot's actuators and view the robot's sensors data
   a) Drive Methods – A reminder of the different robot control methods, including a diagram showing the keyboard's keys that can be used.
   b) Robot Drive – This tab contains the drive controls for the robot, users can change the speed of the robot and change the direction that the robot is going by clicking on the buttons F, B, L, R and Stop to respectively go forward, backward, left, right and stop.
      i) Note: It is possible to use the keys W, A, S, D of the keyboard or a gamepad.
   c) Macro Control – User can disable the controls of the robot's motors.
   d) Battery – User can see the battery voltage and percentage of battery left
   e) Misc. Data – User can view some sensors data including front and rear infrared sensor; accelerometer, gyroscope and magnetometer from an IMU.
   f) Bumpers – User can view the state of the bumpers, to know if the robot is bumping into an object
   g) Pan / Tilt Control – Mouse control for the pan and tilt of a pan tilt camera
   h) Odometry Data – show the linear and angular speed of the robot. This speed is an estimate and can't be taken as accurate.
   i) Front Camera – A camera section where you can view the image coming from the different cameras of the robot.
   j) Rear Camera – A camera section where you can view the image coming from the different cameras of the robot.
   k) Rangefinder – You can visualize the robot environment using either the infrared data or the laser range finder.
   l) Depth Sensor - RGB – If you have a Kinect sensor it is also possible to view the RGB
   m) Depth Sensor - Depth – If you have a Kinect sensor it is also possible to view the Depth image here.
   n) Arm – User can can grab the gripper of the arm by clicking on the yellow point corresponding to it and move the arm as your mouth move. Another widget is present to rotate the base of the arm, available only on certain arms.
   o) Wrist and Gripper – User can change the orientation of the wrist by moving the red bar indicating the orientation of the wrist. User can also choose to open or close the gripper. Note that you can use the keys I,J,K,L to move the arm and the space key to change the state of the gripper as wheel as use a gamepad.
3) Easy control tab – This tab contains only the front camera visualization, the Robot drive section to change the speed and control the robot with the mouse and a section to visualize the map of the environment (which requires a Laser Range Finder and some configuration to display this map)
4) GPS tab – User can visualize the latitude and longitude of the robot. The rest has been deactivated at the moment because of some Google Maps API licensing rights.

2.4.1. Direct Robot Controls

Control the robot from the robot is a very similar procedure as controlling from a computer, but by skipping the step to connect the robot and the computer together. The only thing you have to do is executing the command “roslaunch corobot_teleop corobot_ros_gui.launch” and click on the button connect in the opened window.

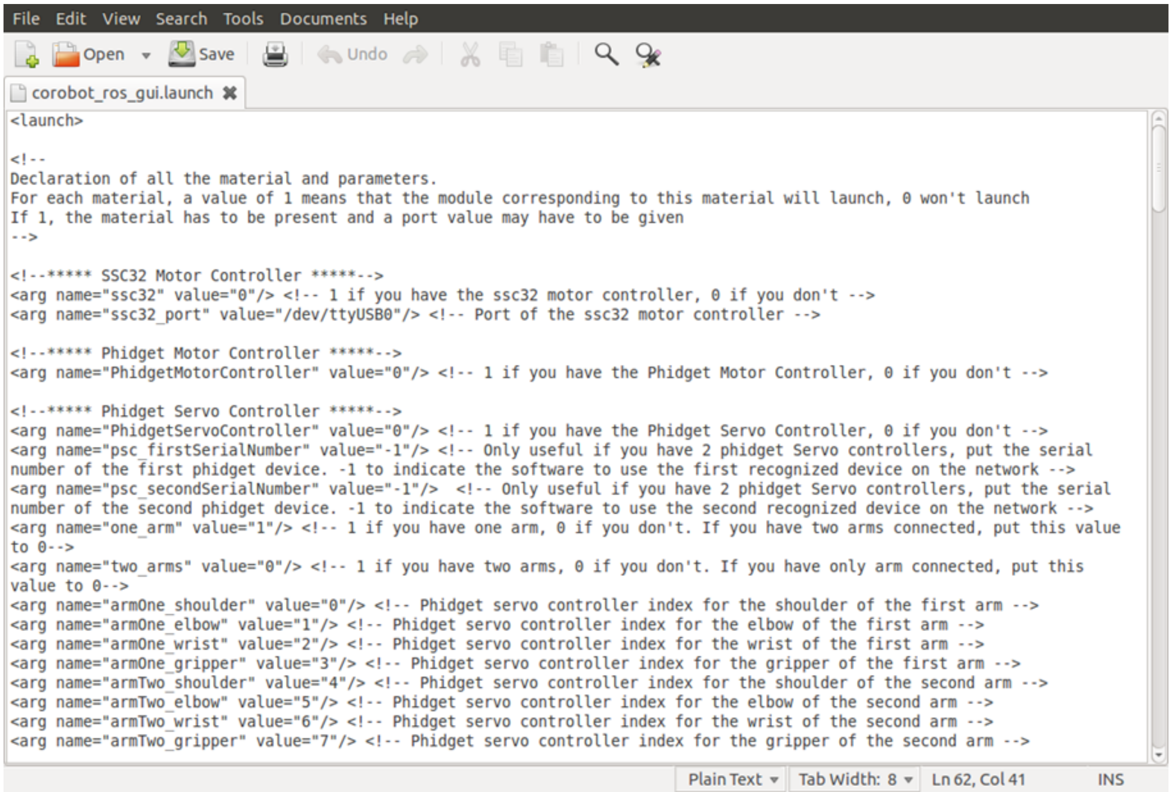
2.4.2. Remote Control Configuration

The robotic operating system uses multiple ‘nodes’ that communicate with each other to control the robot. In order to synchronize the launching of these nodes, ROS utilizes a special configuration file called a launch file. CoroWare’s launch file can be found in the CoroBot_teleop package on the robot and on the computer, in:

~/ros_packages/Corobot/corobot_teleop/launch/corobot_ros_gui.launch

2.4.2.1. Configuration Steps:

Using any text editor, you can configure the launch file based on your robot’s configuration.



```
<launch>

<!--
Declaration of all the material and parameters.
For each material, a value of 1 means that the module corresponding to this material will launch, 0 won't launch
If 1, the material has to be present and a port value may have to be given
-->

<!--***** SSC32 Motor Controller *****-->
<arg name="ssc32" value="0"/> <!-- 1 if you have the ssc32 motor controller, 0 if you don't -->
<arg name="ssc32_port" value="/dev/ttyUSB0"/> <!-- Port of the ssc32 motor controller -->

<!--***** Phidget Motor Controller *****-->
<arg name="PhidgetMotorController" value="0"/> <!-- 1 if you have the Phidget Motor Controller, 0 if you don't -->

<!--***** Phidget Servo Controller *****-->
<arg name="PhidgetServoController" value="0"/> <!-- 1 if you have the Phidget Servo Controller, 0 if you don't -->
<arg name="psc_firstSerialNumber" value="-1"/> <!-- Only useful if you have 2 phidget Servo controllers, put the serial
number of the first phidget device. -1 to indicate the software to use the first recognized device on the network -->
<arg name="psc_secondSerialNumber" value="-1"/>  <!-- Only useful if you have 2 phidget Servo controllers, put the serial
number of the second phidget device. -1 to indicate the software to use the second recognized device on the network -->
<arg name="one_arm" value="1"/> <!-- 1 if you have one arm, 0 if you don't. If you have two arms connected, put this value
to 0-->
<arg name="two_arms" value="0"/> <!-- 1 if you have two arms, 0 if you don't. If you have only arm connected, put this
value to 0-->
<arg name="armOne_shoulder" value="0"/> <!-- Phidget servo controller index for the shoulder of the first arm -->
<arg name="armOne_elbow" value="1"/> <!-- Phidget servo controller index for the elbow of the first arm -->
<arg name="armOne_wrist" value="2"/> <!-- Phidget servo controller index for the wrist of the first arm -->
<arg name="armOne_gripper" value="3"/> <!-- Phidget servo controller index for the gripper of the first arm -->
<arg name="armTwo_shoulder" value="4"/> <!-- Phidget servo controller index for the shoulder of the second arm -->
<arg name="armTwo_elbow" value="5"/> <!-- Phidget servo controller index for the elbow of the second arm -->
<arg name="armTwo_wrist" value="6"/> <!-- Phidget servo controller index for the wrist of the second arm -->
<arg name="armTwo_gripper" value="7"/> <!-- Phidget servo controller index for the gripper of the second arm -->
```

Figure 13 - Text Editor

1) In the launch file, locate the sections for all of the optional equipment offered by CoroWare.
2) Set a value of 1 to items on the robot, and set items that are not present to 0.
   a) Note: This nullifies start nodes for hardware the user does not have. There are no negative effects to having unnecessary nodes running.
   b) Note: You can only change the variables on the robot.

i) A small number of variables need to be modified on the remote computer too, so it is recommended to have to same launch file both on the robot and on the computer, to avoid any problems while controlling the robot remotely.