

## Installing Ubuntu 10.10 on Darwin OP

1. Go to <http://www.ubuntu.com/download/ubuntu/download> and download Ubuntu version 10.10.
2. Create a USB stick out of the downloaded .iso file--follow the directions provided on the website.
3. Connect keyboard, mouse, and monitor to darwin OP.
4. Put USB boot stick into the darwin OP.
5. Turn on robot.
6. While booting, press F2 to enter BIOS menu, then select the "boot" menu, and move 'USB key' to the top of the menu.
7. Restart robot.
8. Install Ubuntu 10.10 following on-screen prompts. Make the username and password something easy to remember, if you are working in a large group. We suggest 'darwin' as a username for each robot. Be sure you name each robot's computer something different though, if you plan to network them together.
9. Restart robot.
10. While booting, press F2 to enter BIOS menu, then select the "boot" menu, and move "USB HDD" to the top of the menu. You may now remove the USB key with the Ubuntu distribution from the robot.
11. Boot into robot. If asked, select the top option to load main Ubuntu partition.
12. To save some space on the robot's hard drive, go to Applications-->Ubuntu Software Center-->Installed Software and remove each instance of an openoffice application.
  - o To accomplish this faster, run 'sudo apt-get remove openoffice.org-common' in a terminal.
13. Now we must make sure the robot can log in automatically. Go to System-->Administration-->Users and Groups. Click 'change' next to the 'Password' option and check the 'Don't ask for password on login' box. Go to System-->Administration-->Login Screen. Click 'unlock'. Uncheck 'Play Login Sound' and select the 'log in as <username>' option. Click 'close'.
14. To disable the startup sound, go to System-->Preferences-->Startup Applications. Uncheck GNOME Login Sound.

## Changing Darwin's IP and setting up Secure Shell

1. Go to System-->Preferences-->Network Connections.
2. Edit Auto eth0.
3. Rename Auto eth0 to 'eth0 dhcp' and uncheck the 'connect automatically' box.
4. Add a new wired connection called 'auto darwinop'.
5. Check the 'Connect Automatically' check box if it is not selected already.
6. Move to the IPv4 settings tab and add a new address. Use 'Manual' as the connection method.
7. Use 192.168.123.1 as your address, 255.255.255.0 as the netmask and 0.0.0.0 as the gateway (or your own settings, if you prefer).
8. Enable internet access on your Darwin OP. This can be done by connecting to the internet wirelessly or through an ethernet cable. We recommend wireless if possible.

9. Open up a terminal and type 'sudo apt-get install openssh-server'. Agree to install the package.
10. After everything has been downloaded and installed, you have completed the setup of IP and Secure shell.

#### Connecting to the robot via SSH using Ubuntu on Main Machine

1. Run an ethernet cable from the robot to your main machine.
2. Boot into Ubuntu on your main machine.
3. After boot, launch a terminal. If you are using a version of Ubuntu which is 10.04 or later, do this using the key combination ctrl+alt+t.
4. Set your IP to be consistent with the IP you set for the robot. If you used the IP suggested here, set it to 192.168.123.10.
  - o `sudo ifconfig eth0 192.168.123.10`
5. To ssh into the robot, use an ssh command from your terminal. If you used the IP suggested above, connect to `darwin@192.168.123.1`
  - o `ssh darwin@192.168.123.1`
6. Congratulations; you are in the robot!

#### Connecting to the robot via VNC using Ubuntu on Main Machine

1. Open the System->Preferences->Remote Desktop
2. Set the following"
  - o "Allow other users to view your desktop"
  - o "Allow other users to control your desktop"
  - o "Require the user to enter this password: " + password
  - o "Configure network automatically to accept connections"
3. Unset the others
4. Now, to ensure that the keyring message never pops up again, do the following:
  - o In a terminal, type '`rm ~/.gnome2/keyrings/login.keyring`'
  - o Reboot
  - o Enter the "System->Preferences->Remote Desktop" dialog
  - o Start entering the password again
  - o In the pop up window for the keyring password set, use a blank password, and accept the resulting "unsafe storage" message
  - o continue typing your password, and close the Remote Desktop dialog.
  - o Reboot

#### Starting the Robot Without a Monitor Attached

1. Open a terminal and navigate to `/etc/default/`.
  - o `cd /etc/default`
2. Edit grub. Make sure you give yourself root permissions to do so.
  - o `sudo nano grub`

3. Change GRUB\_CMDLINE\_LINUX\_DEFAULT="quiet splash" to GRUB\_CMDLINE\_LINUX\_DEFAULT="quiet splash nomodeset". Save and exit the editor.
  - ctrl-o then enter
  - ctrl-x
4. Run update-grub
  - sudo update-grub
5. Create an xorg.conf file to handle the case of there not being a monitor. Place this file in /etc/X11
  - cd /etc/X11
  - sudo nano xorg.conf
  - Type the following into the xorg.conf file:
    - Section "Monitor"
      - Identifier "Monitor0"
      - HorizSync 31-81
      - VertRefresh 56-75
      - EndSection
    - Section "Device"
      - Identifier "Card0"
      - Driver "vesa"
      - EndSection
    - Section "Screen"
      - Identifier "Screen0"
      - Device "Card0"
      - Monitor "Monitor0"
      - EndSection
6. Reboot the robot.

### Getting the UPennalizers Code Release up and Running

While the following steps can be done on the robot directly, it may be easier to download packages on your main machine and then transfer them to the robot via secure copy. The following directions assume you choose to accomplish this via the former method, using secure copy (scp).

#### Install Packages and Dependencies

1. On the robot, run the following line in a terminal.
  - sudo apt-get install build-essential wget subversion cmake swig libreadline6-dev g++ lua5.1

#### Downloading and Installing the Boost Libraries

1. Download the latest version of boost from [www.boost.org](http://www.boost.org).
2. Copy the archive onto the robot.
  - scp boost\_1\_47\_0.tar.gz darwin@192.168.123.1:~/Desktop/

3. Make sure you are in the correct directory of the Darwin (the Desktop)
  - `cd ~/Desktop`
4. Extract boost on the darwin.
  - `tar -xvf boost_1_47_0.tar.gz`
5. Move the boost folder to the /usr/local/ directory
  - `sudo mv boost_1_47_0 /usr/local/`
6. Change to the /usr/local/include directory and make a sym link to the boost files.
  - `cd /usr/local/include`
  - `sudo ln -s /usr/local/boost_1_47_0/boost boost`
7. After this, feel free to delete the tarball from the desktop.
  - `cd ~/Desktop`
  - `rm boost_1_47_0.tar.gz`

### Downloading and Installing Lua Header Files/Dependencies

1. Run the following to get lua 5.1 from the Lua website
  - `wget http://www.lua.org/ftp/lua-5.1.4.tar.gz`
2. Navigate to the directory in which lua is saved and run the following
  - `tar -xvf lua-5.1.4.tar.gz`
3. Change to the lua directory
  - `cd lua-5.1.4`
4. Run the necessary 'make' commands
  - `make linux`
  - `sudo make install`

### Installing the Open-Source Code

1. Download the latest version of the DARwIn open-source release at <https://fling.seas.upenn.edu/~robocup/wiki/>.
2. Copy the tarball to the robot.
  - `scp nsl_open.tar.gz darwin@192.168.123.1:~/Desktop`
3. Navigate to the directory in which the code base is stored and run the following:
  - `tar -xvf nsl_open.tar.gz`
4. Enter the 'Lib' directory of the open-source directory
  - `cd nsl_open`
5. Run the following:
  - `make clean`
  - `make setup_op`
6. Run the following:
  - `sudo rm /lib/udev/rules.d/95-upower-wup.rules` (Or comment out all lines in it - that's what we did)
7. Reboot the robot.

### Running the Open-Source Code

1. Navigate to the Lib directory of the open source code.
  - `cd ../nsi_open/Lib`
2. Run the following make command:
  - `make setup_op`
3. Navigate to the Player/Lib directory
  - `cd ../nsi_open/Player/Lib`
4. Create a screen from which to run dcm.
  - `screen -S dcm`
5. Execute the following command:
  - `lua run_dcm.lua`
6. Detach the dcm screen.
  - `ctrl-A ctrl-D` in quick succession.
7. Navigate to the Player directory
  - `cd ../nsi_open/Player`
8. Create a screen to control the robot.
  - `screen -S player`
9. Execute a lua script to control the robot. This can be any of `test_vision.lua`, `player.lua`, `test_walk.lua`, etc. In this way, you can also create your own lua scripts.
  - Ex: `lua player.lua`
10. If you are running player, detach the screen by repeating step 6. Otherwise, continue to step 11.
11. If you are running a test script, you can now control the robot via the terminal if your script is capable of doing so. If you are running our supplied player code, you can now exit the ssh session and detach the ethernet cable from the robot. To change states, press the middle button on the DARwIn's back.

### Powering off the Robot

1. Press the right-button on the DARwIn's back to kill power to the DARwIn's motors.
2. Press the 'hidden' power button on the DARwIn's chest three times.
3. Wait for the green light on the DARwIn's back to flash five times.
4. Turn off the power.

### Running the Open Source with Webots

1. Change disc to the Lib directory of the open source and run the following commands.
  - `cd ../nsi_open/Lib`
  - `make clean`
  - `make setup_webots_op`
2. Download the nsi\_open webots world code from <https://fling.seas.upenn.edu/~robocup/wiki/>.
3. Change disc to the directory in which you stored the tarball and run the following:
  - `tar -xvf nsi_open_webots.tar.gz`

4. Move the open source directory to the webots project directory.
  - `mv nsl_open_webots /usr/local/webots/projects`
5. Change to the controllers directory of the open source release and execute the following:
  - `cd /usr/local/webots/projects/nsl_open_webots/controllers`
  - `ln -s ../nsl_open/WebotsController dop_lua`
6. Open webots.
7. Open the op1.wbt file in the `/usr/local/webots/projects/nsl_open_webots/worlds` directory.
8. Press play.