MIRO

bioMimetic Robot

EMARO LAB

SETUP FOR MIRO

ABOUT MIRO:
HTTP://CONSEQUENTIALROBOTICS.COM/MIRO/

NOTE

- **The following slides are a summarization of what is described in detail here: https://consequential.bitbucket.io/**
- **Specifically the Developer Manual: https://consequential.bitbucket.io/Developer.html**

CRITICAL POINTS

HARDWARE

- While removing or inserting batteries. Place MiRo as can be seen in the image on the left.
- Quick look at this page (related to basic safety): https://consequential.bitbucket.io/Demonstrator_Commissioning_Before_You_Start.html
- Quick look at this page (if any technical doubts): https://consequential.bitbucket.io/Technical.html

SOFTWARE

- If you wish to use MIROsim (http://consequentialrobotics.com/mirosim/). It is better to have ubuntu as dualboot on your workstation rather than having ubuntu in virtual machine.
- Nonetheless, If you are using virtual machine and MIROsim does not work properly, try again by disabling 3D acceleration (In VirtualMachine settings)

ROS INSTALLATION

Copy and paste in terminal, one by one, everything after the '$' sign.

$ sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'

$ sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116

$ sudo apt-get update

$ sudo apt-get install ros-kinetic-desktop-full

$ sudo rosdep init

$ rosdep update

$ echo "source /opt/ros/kinetic/setup.bash" >> ~/.bashrc

$ source ~/.bashrc

$ sudo apt-get install python-rosinstall python-rosinstall-generator python-wstool build-essential

$ rosversion ros

DOWNLOAD AND PREPARE MIRO DEV KIT AND ANDROID APP

Copy and paste in terminal, everything after the '$' sign.

- **Download MDK (Complete Edition) onto your workstation**
- **Download MIROapp onto your Android mobile device directly**

From: http://labs.consequentialrobotics.com/miro/mdk/

In your workstation:

- **Make a new directory called 'lib' in your 'home' directory**
- **Extract the recently downloaded MDK (Complete Edition) in 'lib', then do this,**

```
$ ln -s ~/lib/mdk-170906 ~/mdk
```

CONFIGURING THE WORKSTATION: TO HAVE STATIC IP

Replace '<instruction_stated_inside>' entirely with the instruction stated inside.

In your workstation:

- **In (Network Preferences/TCP-IP), change (configure IPV4) to 'Manual' from 'DHCP'**
- **Change IPV4 Address to 130.251.13.<select_any_number_between_87_and_94>**
- **In (Network Preferences/DNS), add 130.251.1.4 and 8.8.8.8**
- **If using a virtual machine with ubuntu, all previous steps must be done on the main OS and after which you must go into the network settings of virtual machine and select "attached to" as 'Bridged Adapter'**

CONTINUED...

Copy and paste in terminal, everything after the '$' sign.

Replace '<instruction_stated_inside>' entirely with the instruction stated inside.

- **In your workstation (ubuntu):**

$ ifconfig

- **Check the IP address next to inet addr for enp___**
- **Copy that IP address (It is now the static IP of your workstation)**

$ sudo nano ~/.bashrc

- **Copy and paste in .bashrc the lines below and modify '<>' parts**

```
# cofiguration
export MIRO_PATH_MDK=~/mdk
export ROS_IP=<Put_Here_StaticIP_Of_WorkStation>
export ROS_MASTER_URI=http://localhost:11311

# make our custom messages available to ROS/python
export ROS_PACKAGE_PATH=$MIRO_PATH_MDK/share:$ROS_PACKAGE_PATH
export PYTHONPATH=$MIRO_PATH_MDK/share:$PYTHONPATH

# usual Gazebo setup
source /usr/share/gazebo/setup.sh

# announce MIRO resources to Gazebo
export GAZEBO_RESOURCE_PATH=$MIRO_PATH_MDK/share:${GAZEBO_RESOURCE_PATH}
```

- **Finally**

$ roscore

CONFIGURING MIRO: SO THAT MIRO CAN LOCATE ROSMASTER (YOUR WORKSTATION)

Copy and paste in terminal, everything after the '$' sign.

Replace '<instruction_stated_inside>' entirely with the instruction stated inside.

- **Open MIROapp and check its IP address**
- **In a new terminal on your workstation**

$ ssh root@<Put_Here_IP_Of_MIRO_Found_In_Android_App>

- **MIRO's SSH entry password is**

$ MIROOpen1

- **Now you are inside MIRO's Terminal**

$ sudo nano ~/.profile

- **Find and modify '<>' part related to ROS_IP in .profile (Scroll down almost to the bottom). The lines in the file .profile look similar to what is shown below:**

#.. bla bla (line simply for your reference)
#.. bla bla (line simply for your reference)
#.. bla bla (line simply for your reference)
ROS_IP=<Put_Here_Static_IP_(OfYourWorkstation)_ForConnectingWith_ROS_Master> (the main line to modify)

TESTING SAMPLE CODES

(MIRO is going to move, be careful if it is on the table)

In a new terminal:

- **Run sample python code**

Proceed as shown in the link:
https://consequential.bitbucket.io/Developer_Examples_Python_Command-line_Client.html

- **Run sample C++ code**

Proceed as shown in the link:

https://consequential.bitbucket.io/Developer_Examples_C++_Command-line_Client.html

- **Run sample python code (GUI for sensors and actuators of MIRO):**

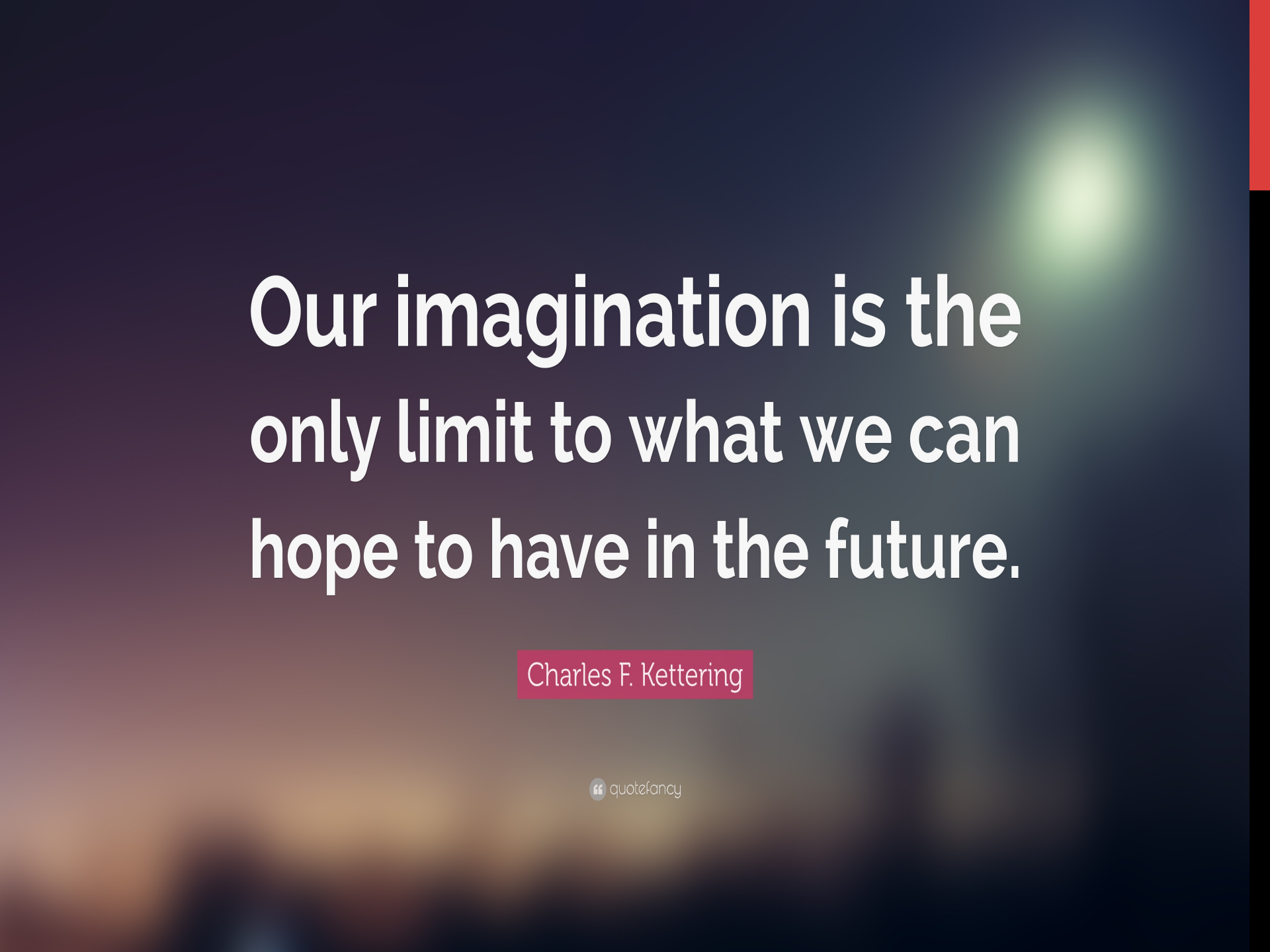
Proceed as shown in the link:

https://consequential.bitbucket.io/Developer_Examples_Python_GUI_Client.html

LAST BUT NOT THE LEAST

- Take a look at Application Domains for Inspiration.

http://consequentialrobotics.com/domains/


Our imagination is the only limit to what we can hope to have in the future.
Charles F. Kettering
quotefancy