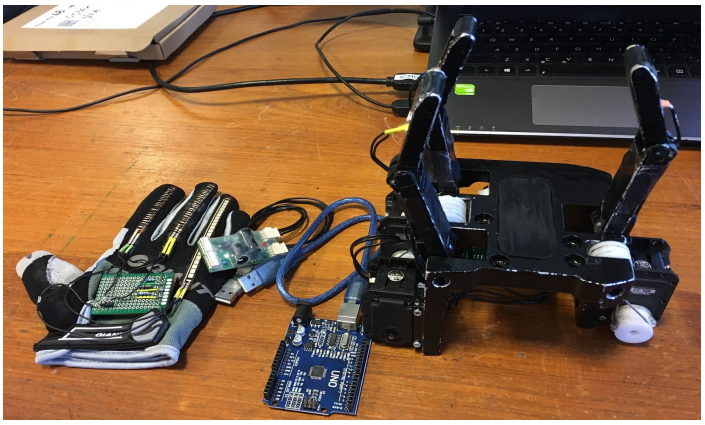
DMT hand and Cyber Glove Instruction Book

Robot intelligence lab
Imperial College London
UROP Students: Kevin(Zhongxuan) Li, Dennis(Siyuan) Dong, Morgan Nightingale
Supervisor Dr Petar Kormushev

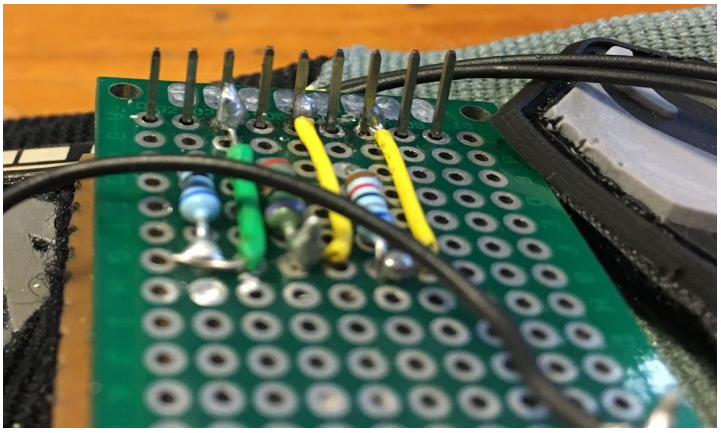
1 Hardware

- Things you need:



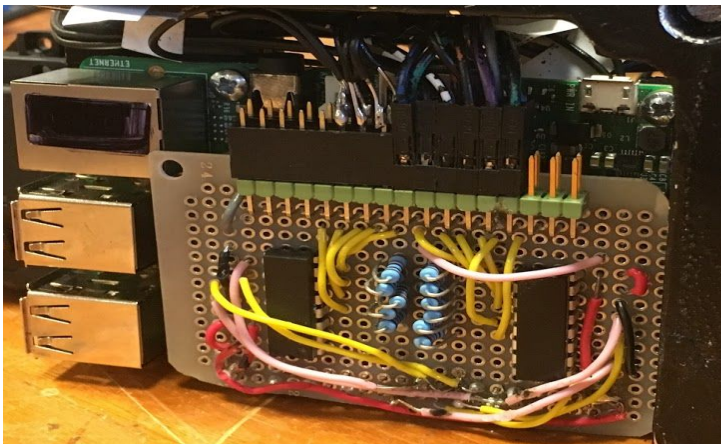
1x arduino uno, 1x motor signal converter, 1x glove, 1x Raspberry Pi(with ROS installed)
In addition, a 12V DC power supply is required.
Raspberry pi is powered by the 5V buck converter on DeNiro.

- Pin configuations of the cyber glove



The leftmost pin(aligned with resistor) is 5V power input, rightmost pin is common ground.
While the middle three pins are analogue outputs from the glove to arduino.

- Pin configurations of the DMT hand

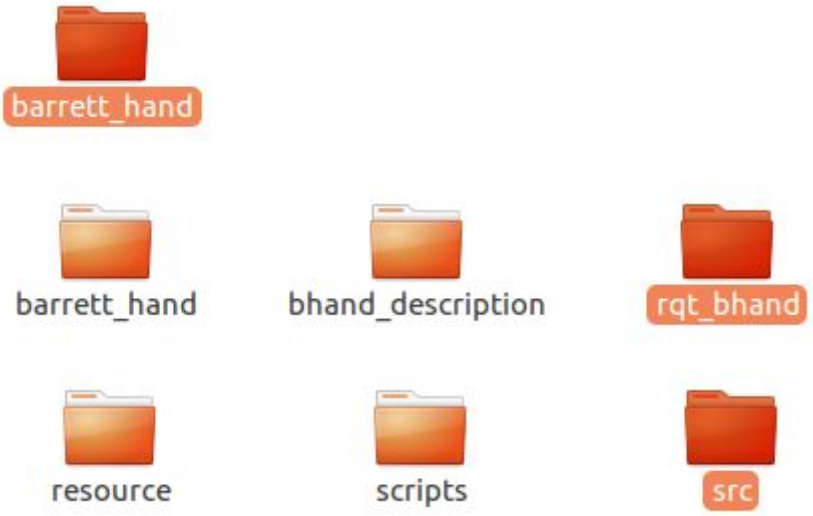


The second row are grounds whereas the first row are connected to GPIO pins of the raspberry pi. Two ADC converters(Adafruit MCP3008) have been used, each has 8 digital output channels as well as CLK, CS, and GROUND.

2 Software

- Source code is available at:
  https://github.com/Dieselmarble/2017urop.git
  I recommend you to recreate the ROS package for GUI, since after git clone, it doesn't work on all PCs. Be aware of the Cmake.list.

  My file hierarchy:

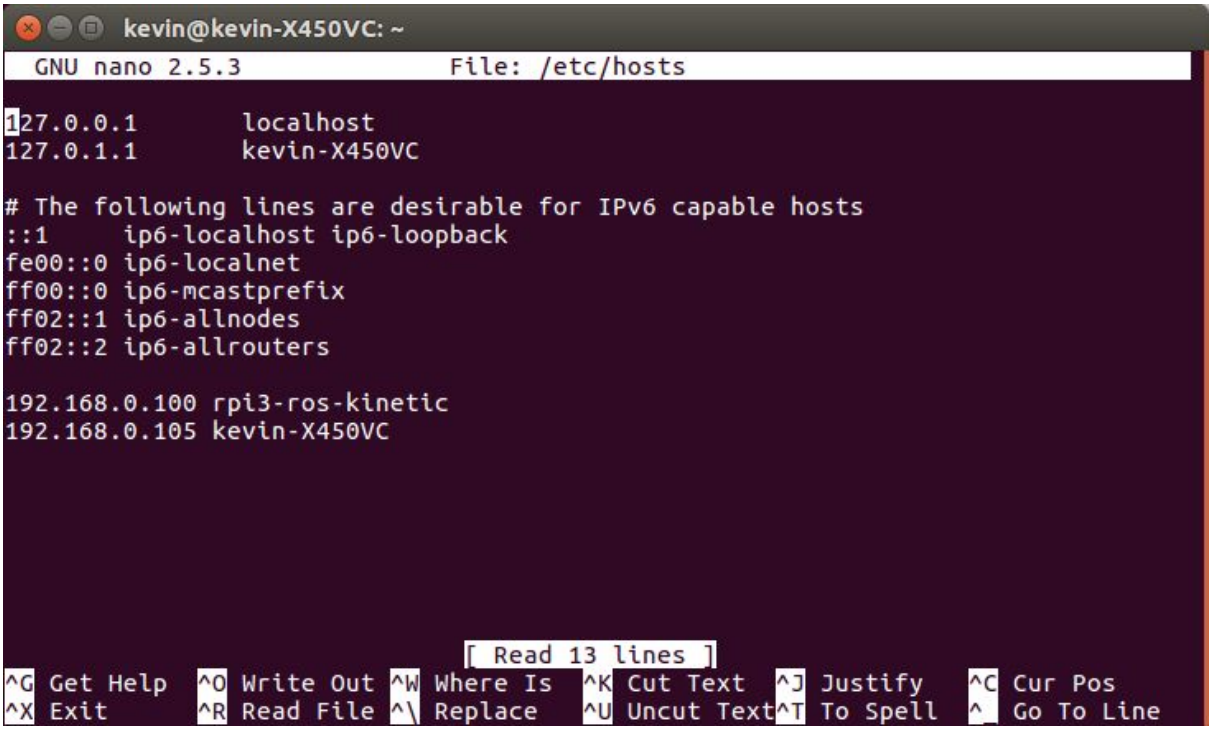


2.1 Launch the controller manager.

- The controller manager is supposed to be launched on the Raspberry pi preassembled on the DMT hand. The Raspberry pi is connected wirelessly to the laptop through SSH. However, as you can see, the HDMI port is blocked by the metal

body thus we cannot export display from the pi, so I recommend using an ethernet cable and draw the pi's display on your laptop.

- Check IP addresses of Raspberry pi and your laptop, in both terminal type: **$ hostname -I** Reference: http://wiki.ros.org/ROS/Tutorials/MultipleMachines
- On the pi's terminal type:
  $ sudo nano /etc/hosts, this may require the pi's password, which is "robotdeniro"
- This command opens your hosts file from which you need type IP address of your laptop on the bottom line.



- Do exactly the same thing on your laptop and type IP address of the raspberry pi on the bottom line of the pc's hosts file.
- Note IP address changes every time you restart your device and you need to follow the instructions again.

2.2 Start your roscore

- (Optional) On laptop: **$ ssh rpi3-ros-kinetic**
  this will log you into the pi's terminal and make life easier.
  Be aware that Raspberry pi's hostnames may change.

- On the Raspberry pi terminali:
  $ roscore

```
roscore http://kevin-X450VC:11311/
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://kevin-X450VC:38069/
ros_comm version 1.12.7


SUMMARY
========

PARAMETERS
 * /rosdistro: kinetic
 * /rosversion: 1.12.7

NODES

auto-starting new master
process[master]: started with pid [3974]
ROS_MASTER_URI=http://kevin-X450VC:11311/

setting /run_id to cff49d10-8e48-11e7-b7de-48d224c201de
process[rosout-1]: started with pid [3987]
started core service [/rosout]
```

There can only be one ROS master within two devices. It is better to have your ROS master and controller manager running at same side(both on the Raspberry pi). But it is not necessary.

- Plug the “USB2AX” motor signal converter in USB port, this should appears as port name **“dev/tty/ACM0”** check by **$ ls /dev**
- Open the launch file “controller_manager.launch”**.** Type:
  $ roslaunch dynamixel_controllers controller_manager.launch

```
kevin@kevin-X450VC: ~/catkin_ws
kevin@kevin-X450VC:~$ cd catkin_ws
kevin@kevin-X450VC:~/catkin_ws$ source devel/setup.bash
kevin@kevin-X450VC:~/catkin_ws$ roslaunch dynamixel_controllers controller_manag
er.launch
... logging to /home/kevin/.ros/log/cff49d10-8e48-11e7-b7de-48d224c201de/roslaun
ch-kevin-X450VC-5025.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://kevin-X450VC:33683/

SUMMARY
========

PARAMETERS
 * /dynamixel_manager/namespace: dynamixel_control...
 * /dynamixel_manager/serial_ports/dxl_tty1/baud_rate: 57600
 * /dynamixel_manager/serial_ports/dxl_tty1/max_motor_id: 10
 * /dynamixel_manager/serial_ports/dxl_tty1/min_motor_id: 1
 * /dynamixel_manager/serial_ports/dxl_tty1/port_name: /dev/ttyACM0
 * /dynamixel_manager/serial_ports/dxl_tty1/update_rate: 5
 * /rosdistro: kinetic
 * /rosversion: 1.12.7
```

- On laptop :

$ export ROS_MASTER_URI=http://rpi3-ros-kinetic:11311
This sets roscore on raspberry pi your ROS_MASTER.
$ rostopic list to check all rostopic are correctly created.
$ rosservice list to check all rosservice are correctly opened

2.3 Run the GUI

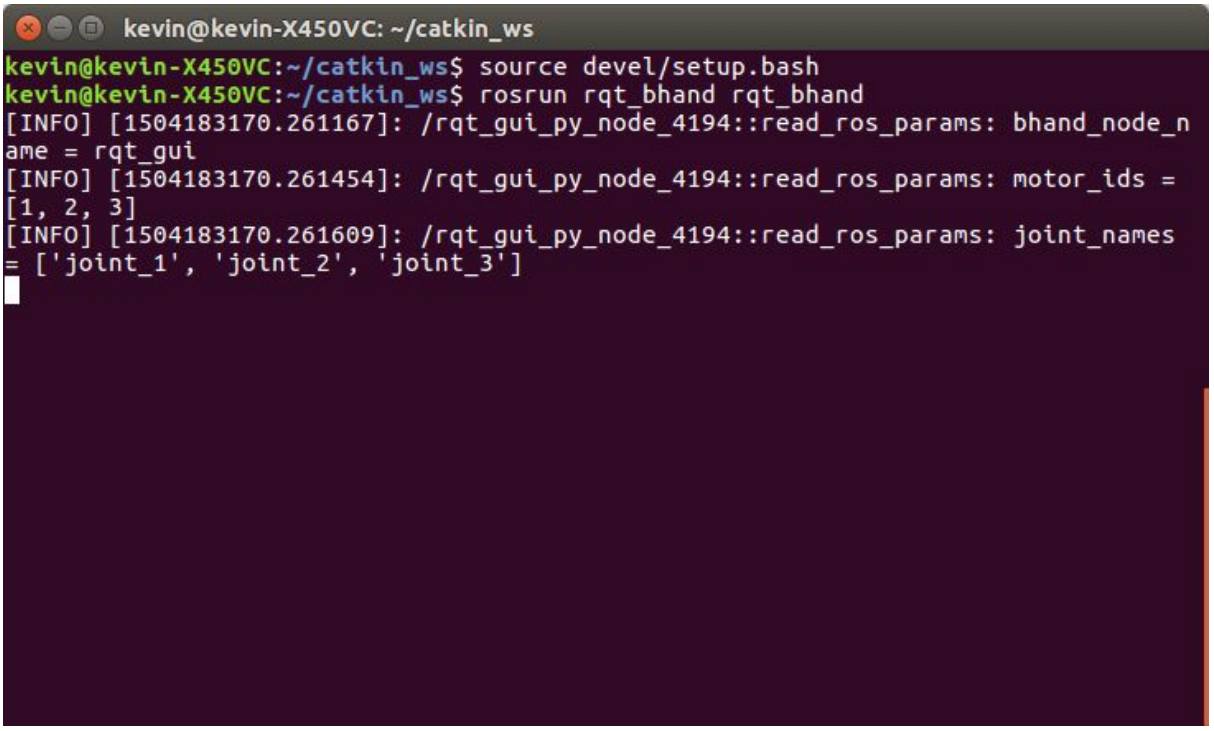
- On laptop:
  $ rosrun rqt_bhand rqt_bhand
- Note if you don't connect the arduino to USB port. Terminal will say:

```
SerialException: [Errno 2] could not open port /dev/ttyUSB0: [Errno 2] No such f
ile or directory: '/dev/ttyUSB0'
```

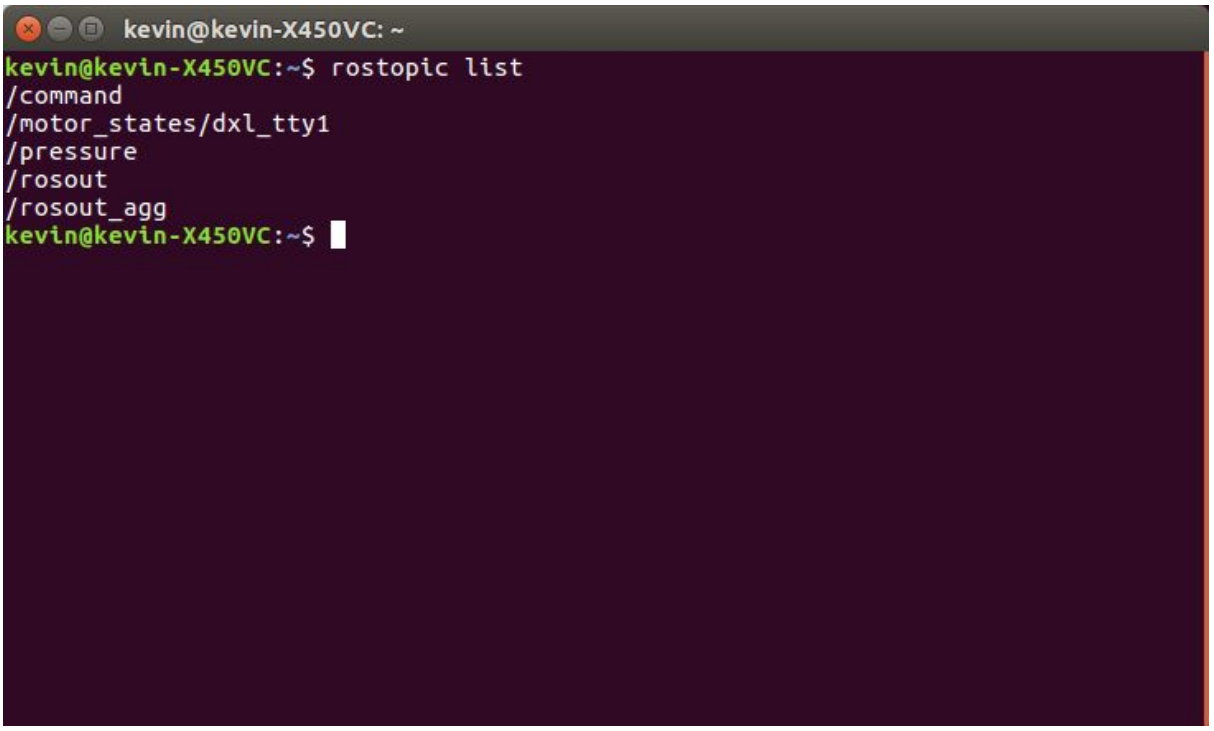
- To solve this, you can either connect the arduino to USB port or just comment the following line in rqt_bhand.py if you don't need the cyber glove,

```
#serial port for sybergove and arduino
ser=serial.Serial("/dev/ttyUSB0", baudrate=9600, timeout=1)
```

- After the GUI is successfully started you will see:

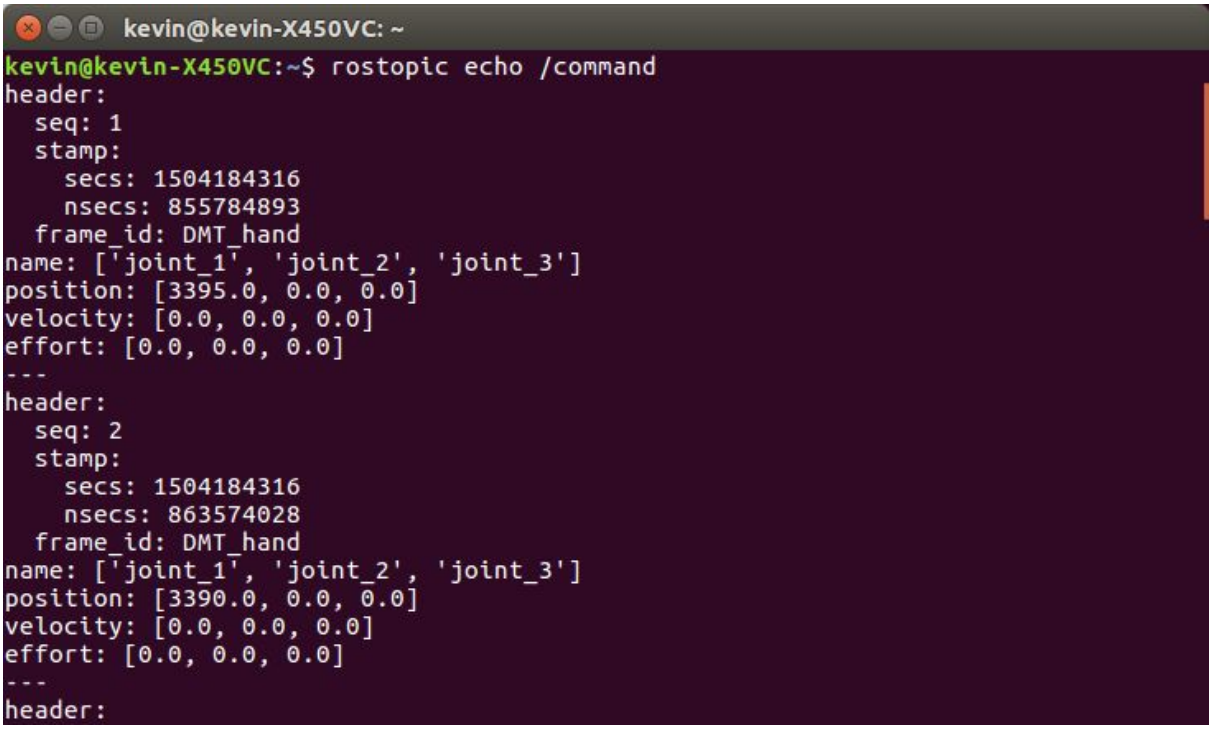


- To exam all rostopics, open a new terminal and type:
  $ rostopic list, after which you will see:



where “/command” is for joint_state control of the hand, “motor_states/dxl_tty1” is for all motor configurations published by the controller_manager and /pressure is for the tactile sensors assembled on three fingers.

- For instance, **$rostopic echo /command** show you the joint_states data the GUI is sending underway.



- About how to use the GUI, see: https://www.youtube.com/watch?v=tklL4wqzF8Q

- If you are using the cyber glove, launch arduino IDE and upload “glove.ino” to the board. The GUI uses Pythonserial to read data drom arduino. Be carefull with String-Float conversion here. Example Code :

```
def getvalues(self):
                                        ser.write(b'y')
                                        arduinoData = format(ser.readline().strip())
                                        if arduinoData.isdigit():
                                                return float(arduinoData)
                                        else:
                                                return 0
```