Note: This tutorial assumes that you have completed the previous tutorials: ROS tutorials (/ROS/Tutorials).

Please ask about problems and questions regarding this tutorial on answers.ros.org (http://answers.ros.org). Don't forget to include in your question the link to this page, the versions of your OS & ROS, and also add appropriate tags.

Configuring and Using a Linux-Supported Joystick with ROS

Description: This tutorial is an introduction to using the joystick connected to a desktop computer. After reading it, you should be able to bring up the joy node and display the data coming from the joystick over ROS.

Keywords: joystick, driver

Tutorial Level: BEGINNER

Next Tutorial: Writing a Teleoperation Node for a Linux-Supported Joystick (/joy/Tutorials/WritingTeleopNode)

Contents

1. Installing
2. Configuring the Joystick
3. Starting the Joy Node

0.1 Installing

Start by installing the package:

```
$ sudo apt-get install ros-indigo-joy
```

0.1 Configuring the Joystick

Connect your joystick to your computer. Now let's see if Linux recognized your joystick.

```
$ ls /dev/input/
```

You will see a listing of all of your input devices similar to below:

```
by-id    event0  event2  event4  event6  event8  mouse0  mouse2  uinput
by-path  event1  event3  event5  event7  js0     mice    mouse1
```

As you can see above, the joystick devices are referred to by jsX ; in this case, our joystick is js0. Let's make sure that the joystick is working.

```
$ sudo jstest /dev/input/jsX
```

You will see the output of the joystick on the screen. Move the joystick around to see the data change.

```
Driver version is 2.1.0.
Joystick (Logitech Logitech Cordless RumblePad 2) has 6 axes (X, Y, Z, R
z, Hat0X, Hat0Y)
and 12 buttons (BtnX, BtnY, BtnZ, BtnTL, BtnTR, BtnTL2, BtnTR2, BtnSelec
t, BtnStart, BtnMode, BtnThumbL, BtnThumbR).
Testing ... (interrupt to exit)
Axes:  0:     0  1:     0  2:     0  3:     0  4:     0  5:     0 Button
s:  0:off  1:off  2:off  3:off  4:off  5:off  6:off  7:off  8:off  9:off
10:off 11:off
```

Now let's make the joystick accessible for the ROS joy node. Start by listing the permissions of the joystick:

```
$ ls -l /dev/input/jsX
```

You will see something similar to:

```
crw-rw-XX- 1 root dialout 188, 0 2009-08-14 12:04 /dev/input/jsX
```

If XX is rw: the js device is configured properly.

If XX is --: the js device is not configured properly and you need to:

```
$ sudo chmod a+rw /dev/input/jsX
```

0.1 Starting the Joy Node

To get the joystick data published over ROS we need to start the joy node. First let's tell the joy node which joystick device to use- the default is js0.

```
$ roscore
$ rosparam set joy_node/dev "/dev/input/jsX"
```

Now we can start the joy node.

```
$ rosrun joy joy_node
```

You will see something similar to:

```
[ INFO] 1253226189.805503000: Started node [/joy], pid [4672], bound on
[aqy], xmlrpc port [33367], tcpros port [58776], logging to [/u/mwise/ro
s/ros/log/joy_4672.log], using [real] time

[ INFO] 1253226189.812270000: Joystick device: /dev/input/js0

[ INFO] 1253226189.812370000: Joystick deadzone: 2000
```

Now in a **new terminal** you can `rostopic echo` the joy topic to see the data from the joystick:

```
$ rostopic echo joy
```

As you move the joystick around, you will see something similar to :

```
---
axes: (0.0, 0.0, 0.0, 0.0)
buttons: (0, 0, 0, 0, 0)
---
axes: (0.0, 0.0, 0.0, 0.12372203916311264)
buttons: (0, 0, 0, 0, 0)
---
axes: (0.0, 0.0, -0.18555253744125366, 0.12372203916311264)
buttons: (0, 0, 0, 0, 0)
---
axes: (0.0, 0.0, -0.18555253744125366, 0.34022033214569092)
buttons: (0, 0, 0, 0, 0)
---
axes: (0.0, 0.0, -0.36082032322883606, 0.34022033214569092)
buttons: (0, 0, 0, 0, 0)
```

On electric (/electric) and above the new sensor_msgs/Joy (http://docs.ros.org/en/api/sensor_msgs/html/msg/Joy.html) message is emitted, which includes a Header (/msg#headerSect):

```
---
header:
  seq: 9414
  stamp:
    secs: 1325530130
    nsecs: 146351623
  frame_id: ''
axes: [-0.0038758506998419762, -0.0038453321903944016, -0.0, -0.999969482
421875, 0.0, 0.0]
buttons: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
---
header:
  seq: 9415
  stamp:
    secs: 1325530130
    nsecs: 146351623
  frame_id: ''
axes: [-0.0038758506998419762, -0.0038453321903944016, -0.0, -0.999969482
421875, 0.0, 0.0]
buttons: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
---
header:
  seq: 9416
  stamp:
    secs: 1325530130
    nsecs: 146351623
  frame_id: ''
axes: [-0.0038758506998419762, -0.0038453321903944016, -0.0, -0.999969482
421875, 0.0, 0.0]
buttons: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
---
```


Wiki: joy/Tutorials/ConfiguringALinuxJoystick (last edited 2016-11-20 06:16:49 by MatthewWilson (/MatthewWilson))