

# ROS序列埠範例

本文檔介紹如何在ROS下來讀取HI226/HI229的數據，並提供了c++語言範例代碼，通過執行ROS命令，運行相應的節點，就可以看到打印到終端上的資訊。

- 測試環境：Ubuntu16.04
- ROS版本：ROS Kinetic Kame
- 測試設備：HI226 HI229

## 1. 安裝USB-UART驅動

Ubuntu 系統自帶CP210x的驅動，默認不需要安裝序列埠驅動。將調試版連接到電腦上時，會自動識別設備。識別成功後，會在dev目錄下出現一個對應的設備ttyUSBx

檢查USB-UART設備是否被Ubuntu識別：

1. 打開命令行(ctrl + alt + t)
2. 輸入 ls /dev 查看是掛載成功USB轉序列埠設備
3. 查看是否存在 ttyUSBx 這個設備文件。x表示USB設備號，由於Ubuntu USB設備號為從零開始依次累加，所以多個設備每次開機後設備號是不固定的，需要確定設備的設備號。下圖為沒有插入HI226/HI229評估板時的dev設備列表，這個時候，dev目錄下並沒有名為 **ttyUSB** 文件

```
linux@ubuntu:~$ cd /dev
linux@ubuntu:/dev$ ls
agpgart      loop3       snapshot    tty33        tty7         ttyS8
autofs       loop4       snd          tty34        tty8         ttyS9
block        loop5       sr0          tty35        tty9         uhid
bsg          loop6       stderr       tty36        ttyprintk    uinput
btrfs-control loop7       stdin        tty37        ttyS0        urandom
bus          loop-control mapper       tty38        ttyS1        userio
cdrom        mcelog     mtd          tty39        ttyS10       vcs
cdrw         mem        memory_bandwidth tty40        ttyS11       vcs1
char         midi        memory_bandwidth tty41        ttyS12       vcs2
console      mqueue     mtd          tty42        ttyS13       vcs3
core         net         mtd          tty43        ttyS14       vcs4
cpu_dma_latency cuse       network_latency tty44        ttyS15       vcs5
disk         dmidevent  network_throughput tty45        ttyS16       vcs6
dmmidi       dri         null         tty46        ttyS17       vcs7
dvd          dvd         port         tty47        ttyS18       vcsa
ecryptfs     fb0         ppp          tty48        ttyS19       vcsa1
fd           fd          psaux        tty49        ttyS20       vcsa2
full         full        ptmx         tty50        ttyS21       vcsa3
fuse         fuse        pts          tty51        ttyS22       vcsa4
hidraw0      hpet        random       tty52        ttyS23       vcsa5
hpet         hugepages  rfkill       tty53        ttyS24       vcsa6
hwrng        initctl     rtc          tty54        ttyS25       vcsa7
input        kmsg       rtc0         tty55        ttyS26       vfio
lightnvmm    log         sda          tty56        ttyS27       vga_arbiter
loop0        loop1       sda1         tty57        ttyS28       vhci
loop1        loop2       sda2         tty58        ttyS29       vhost-net
loop2        shm         sda5         tty59        ttyS30       vhost-vsock
linux@ubuntu:/dev$
```

4. 插入USB線，連接調試板，然後再次執行ls /dev。 dev目錄下多了一個設備，如圖：

```
linux@ubuntu:/dev$ ls
agpgart      loop3      shm        tty32       tty63       ttyS7
autofs       loop4      snapshot   tty33       tty7         ttyS8
block        loop5      snd        tty34       tty8         ttyS9
bsg          loop6      sr0        tty35       tty9         ttyUSB0
btrfs-control loop7      stderr     tty36       ttyprintk    uinput
bus          loop-control stdin       tty37       ttyS0
```

ttyUSB0 文件就是調試版在ubuntu系統中生成的設備(後面的數字是不固定的，有可能為 ttyUSB1 或 ttyUSB2)

5. 打開USB設備的可執行權限：

```
1 $ sudo chmod 777 /dev/ttyUSB0
```

## 2. 安裝ROS serial軟體包

本範例依賴ROS提供的serial包實現序列埠通信。

1. 首先執行如下命令，下載安裝serial軟體包：

```
1 $ sudo apt-get install ros-kinetic-serial
```

2. 然後輸入 `roscd serial` 命令，進入serial下載位置，如果安裝成功，就會出現如下資訊：

```
1 $:/opt/ros/kinetic/share/serial
```

## 3. 編譯serial\_imu\_ws工作空間

1. 打開終端進入/examples/ROS/serial\_imu\_ws 目錄
2. 執行catkin\_make命令，編譯成功後出現完成度100%的資訊。

## 4. 修改序列埠波特率和設備號

1. 在Ubuntu環境中，支援的波特率為115200, 460800, 921600。本範例使用的默認波特率是115200，默認打開的序列埠名稱是/dev/ttyUSB0。
2. 如果您需要更高的輸出頻率，請編輯serial\_imu.cpp文件，修改serial\_imu.cpp文件中的宏定義，改為其他波特率。

```
1 #define IMU_SERIAL "/dev/ttyUSB0"
2 #define BAUD 115200
```

注意修改後需要回到serial\_imu\_ws目錄下，重新執行catkin\_make命令

## 5. 顯示數據

本範例提供了三種查看數據方式：

1. 第一種方式是顯示所有的數據資訊，通過printf把imu上傳的所有的資訊都打印到終端上，便於查看數據。
2. 打印ROS標準imu\_msg 數據
3. rviz工具實現可視化

## 5.1: 輸出IMU原始數據

1. 打開另一個終端，執行 `roscore` 開啟ROS

```
1 $ roscore
```

2. 回到serial\_imu\_ws文件夾下 執行

```
1 $ source devel/setup.bash
```

2. 執行啟動roslaunch 啟動接受程序

```
1 $ roslaunch serial_imu serial_imu
```

執行成功後，就可以看到所有的資訊：

```
1
2   Devie ID:      0
3   Run times: 0 days  3:26:10:468
4   Frame Rate:    100Hz
5   Acc(G):       0.933    0.317    0.248
6   Gyr(deg/s):   -0.02    0.30    -0.00
7   Mag(uT):      0.00     0.00    0.00
8   Eul(R P Y):   52.01   -66.63   -60.77
9   Quat(W X Y Z): 0.770    0.066   -0.611   -0.172
1  Pleaes enter ctrl + 'c' to quit....
0
```

## 5.2: 輸出ROS標準 Imu.msg

1. 在windows系統下進行配置模組，使能四元數輸出。
2. 使用Window下 Uranus上位機進行配置：先把模組連接到PC機上。然後使用Uranus工具進行 連接對應的com口，點擊 **工具 ---> 配置模組**，在協議配置區域，可以選擇老協議中單獨勾選 **加速度、角速度、四元數**，或者是選擇新協議的 **IMU數據集合**。勾選好之後，點擊 **寫入配置**，接收區最後顯示 **ok**，說明配置成功。在關閉配置窗口上，看一下數據顯示區域，最後確認一下，加速度、角速度、四元數是否正確輸出。執行 `roslaunch imu_launch imu_msg.launch` 命令。執行成功後，就可以看到ROS定義的IMU話題消息：

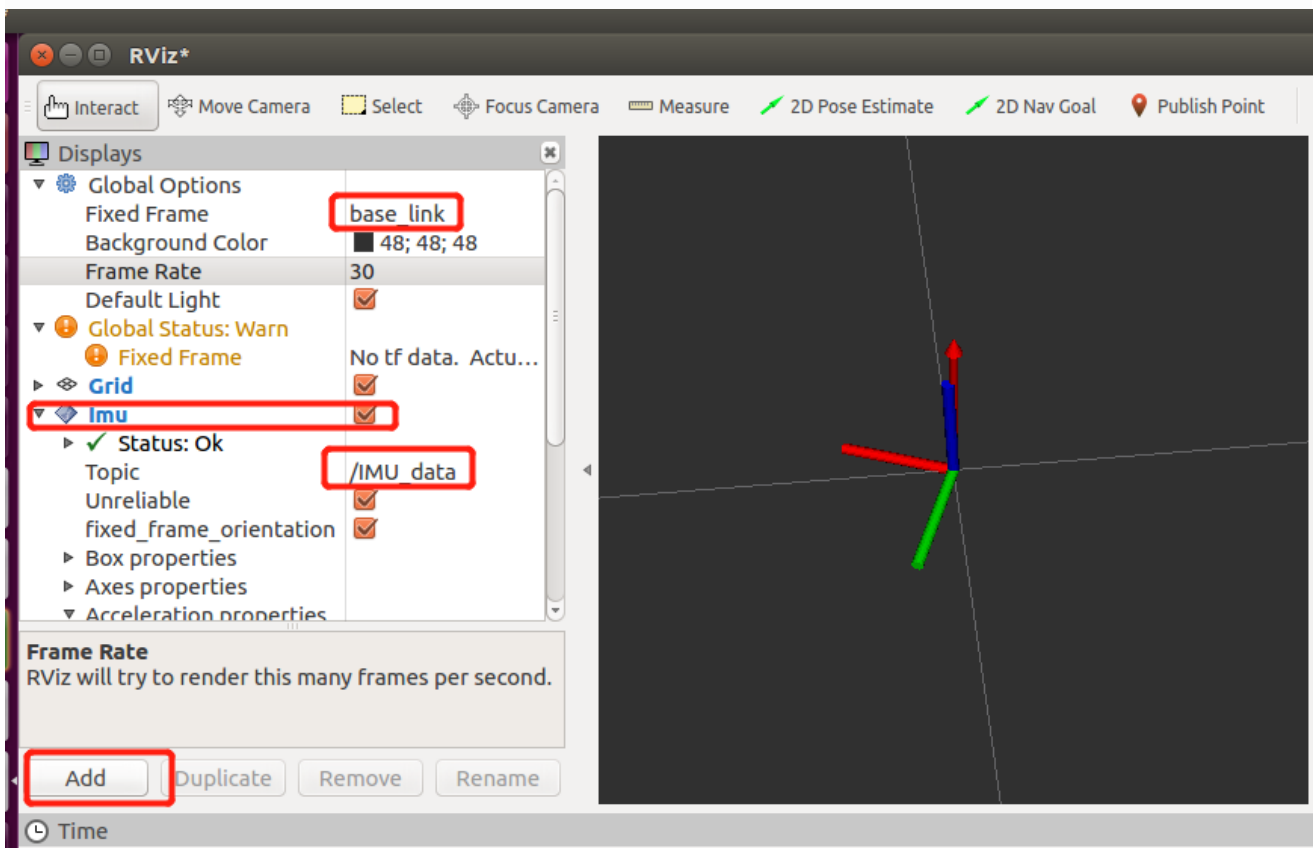
```
1 header:
2   seq: 595
3   stamp:
4     secs: 1595829903
5     nsecs: 680423746
6   frame_id: "base_link"
7 orientation:
8   x: 0.0663746222854
9   y: -0.611194491386
10  z: -0.17232863605
11  w: 0.769635260105
12 orientation_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
13 angular_velocity:
14   x: 0.0851199477911
15   y: 0.0470183677971
16   z: 0.00235567195341
17 angular_velocity_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
18 linear_acceleration:
19   x: 0.93323135376
20   y: 0.317857563496
```

```
21 z: 0.247811317444
22 linear_acceleration_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
23
```

## 5.3: rviz可視化

1. 安裝ROS rviz插件
2. 同上節，使能模組四元數輸出
3. 進入serial\_imu\_ws，執行 `roslaunch imu_launch imu_rviz.launch` 命令，執行成功後，rviz工具被打開。
4. 先點擊左下角的Add標籤，然後在彈出窗口中，選擇 By display type標籤，查找rviz\_imu\_plugin；找到之後，選擇它下面的imu標籤，點擊OK，這時，我們可以看到rviz的左側的展示窗口中已經成功添加上了Imu的標籤。在FixedFrame中填入**base\_link**。topic中添加 **/IMU\_data**。這時，可以看到坐標系隨感測器改變而改變。
5. 可以從這裡下載rviz的工具：

```
1 git clone -b indigo https://github.com/ccny-ros-pkg/imu_tools.git
```



## 6. FAQ

1. 如果在執行 `roslaunch serial_imu serial_imu` 時候，出現如下錯誤：

```
linux@ubuntu:~$ roslaunch serial_imu serial_imu
[rospack] Error: package 'serial_imu' not found
```

這是由於沒有配置環境的原因導致的，解決辦法就是在當前終端執行 `source ~/serial_imu_ws/devel/setup.bash` 命令。但是這個辦法並不能一次性解決，每次開啟一個終端，運行新節點都需要為該終端設置環境變量。所以按照如下方式，可以不用這麼麻煩：執行 `gedit ~/.bashrc` 命令，打開一個文件，然後在這個文件的末尾加入ROS程序註冊命令。(serial\_imu\_ws\_dir為serial\_imu\_ws所在目錄)

```
1 $ source <serial_imu_ws_dir>/devel/setup.bash
```

2. 序列埠打開失敗，權限不夠。執行chmod命令，開啟權限。

```
1 $ sudo chmod 777 /dev/ttyUSB0
```