

DAISCH IM1R

专为机器人而生的 IMU

了解更多



IM1R_ROS_Driver

issues 1 open

pull requests 0 open

license BSD-3-Clause

release v1.1.0

stars 6

[English](#) · [简体中文](#) · [Global Official Site](#) · [中文官网](#)

目录

- [IM1R ROS Driver](#)
 - [目录](#)
 - [项目描述](#)
 - [入门指南](#)
 - [系统要求](#)
 - [安装步骤](#)
 - [使用说明](#)
 - [发布的话题](#)
 - [参数介绍](#)
 - [标准话题](#)
 - [imu/data](#)
 - [temperature](#)
 - [自定义话题](#)
 - [im1r/extra](#)
 - [IM1R配置工具](#)
 - [启动工具](#)
 - [更改配置项](#)
 - [注意事项](#)
 - [常见问题解答](#)
 - [问题1: 找不到脚本](#)
 - [贡献](#)

- [许可证](#)

项目描述

本项目旨在开发和维护适用于 [IM1R-FB-U](#) 的 ROS 驱动程序。[IM1R-FB-U](#) 是一款由 [DAISCH](#) 设计的紧凑型 6 轴惯性测量模块。

入门指南

系统要求

- Ubuntu 18.04 / ROS Melodic
- Ubuntu 20.04 / ROS Noetic

安装步骤

1. 安装 ROS:

请参考 [ROS 安装指南](#) 获取详细说明。

2. 安装依赖项:

运行以下命令安装依赖项:

- 对于Ubuntu 18.04 / Python 2

```
sudo apt update
sudo apt install python-pip
pip install pyserial
```

- 对于Ubuntu 20.04 / Python 3

```
sudo apt update
sudo apt install python3-pip
pip3 install pyserial
```

3. 创建 catkin 工作空间:

```
mkdir -p ~/catkin_ws/src
```

4. 克隆项目仓库到 src 目录:

```
cd ~/catkin_ws/src
git clone https://github.com/DAISCHSensor/IM1R_ROS_Driver.git
```

- 对于Ubuntu 20.04 / Python 3, 需要切换到适配的分支

```
cd IM1R_ROS_Driver
git checkout ubuntu20.04-support
```

5. 构建工作空间:

```
cd ~/catkin_ws/
catkin_make
```

6. 添加工作空间的环境变量到 `.bashrc`:

```
echo "source ~/catkin_ws/devel/setup.bash" >> ~/.bashrc
source ~/.bashrc
```

使用说明

1. 通过UART1数据线束连接IM1R

2. 启动 ROS 核心服务:

```
roscore
```

3. 识别 IM1R 设备的串口:

```
dmesg | grep tty
```

4. 设置串口权限:

假设 IM1R 设备连接到 `/dev/ttyUSB0`:

```
sudo chmod 666 /dev/ttyUSB0
```

5. 启动驱动节点:

```
roslaunch im1r_ros_driver daisch_im1r_node.py /dev/ttyUSB0 115200
```

- `/dev/ttyUSB0` 是当前IM1R连接的串口
- `115200` 是当前IM1R使用的波特率

6. 列出所有话题:

```
rostopic list
```

7. 输出指定话题的内容:

```
rostopic echo imu/data
```

8. (示例程序)订阅话题:

```
roslaunch im1r_ros_driver subscriber_example.py
```

发布的话题

- `imu/data` ([sensor_msgs/Imu](#)) 四元数、角速度和线性加速度
- `temperature` ([sensor_msgs/Temperature](#)) 来自设备的温度
- `im1r/extra` ([DAISCH 自定义话题](#)) 来自 **IM1R** 的额外参数

参数介绍

标准话题

imu/data

Variable	Supported
time <code>header.stamp</code>	✓
string <code>header.frame_id</code>	✓
float64 <code>orientation.x</code>	✓
float64 <code>orientation.y</code>	✓
float64 <code>orientation.z</code>	✓
float64 <code>orientation.w</code>	✓
float64[9] <code>orientation_covariance</code>	X
float64 <code>angular_velocity.x</code>	✓
float64 <code>angular_velocity.y</code>	✓
float64 <code>angular_velocity.z</code>	✓
float64[9] <code>angular_velocity_covariance</code>	X
float64 <code>linear_acceleration.x</code>	✓
float64 <code>linear_acceleration.y</code>	✓
float64 <code>linear_acceleration.z</code>	✓
float64[9] <code>linear_acceleration_covariance</code>	X

temperature

Variable	Supported
time <code>header.stamp</code>	✓
string <code>header.frame_id</code>	✓
float64 <code>temperature</code>	✓
float64 <code>variance</code>	X

自定义话题

im1r/extra

Variable	Type	Definition	Unit	Remarks
count	uint8	Message counter	-	0~255 cyclic increment
timestamp	uint64	Timestamp of the measurement	microseconds (μs)	UNIX time
pitch	float64	Pitch angle	degrees (°)	
roll	float64	Roll angle	degrees (°)	
yaw	float64	Yaw angle	degrees (°)	
imu_status	uint8	IMU status indicator	-	Bit 0: Acceleration valid (0) / invalid (1) Bit 2: Angular velocity valid (0) / invalid (1) Higher bits are not defined
gyro_bias_x	float64	Gyroscope bias along the X axis	radians/second (rad/s)	
gyro_bias_y	float64	Gyroscope bias along the Y axis	radians/second (rad/s)	
gyro_bias_z	float64	Gyroscope bias along the Z axis	radians/second (rad/s)	
gyro_static_bias_x	float64	Static gyroscope bias along the X axis	radians/second (rad/s)	
gyro_static_bias_y	float64	Static gyroscope bias along the Y axis	radians/second (rad/s)	
gyro_static_bias_z	float64	Static gyroscope bias along the Z axis	radians/second (rad/s)	

IM1R配置工具

启动工具

1. 通过UART2调试线束连接IM1R

2. 启动 ROS 核心服务：

```
roscore
```

3. 识别 IM1R 设备的串口：

```
dmesg | grep tty
```

4. 设置串口权限：

假设 IM1R 设备连接到 /dev/ttyUSB0：

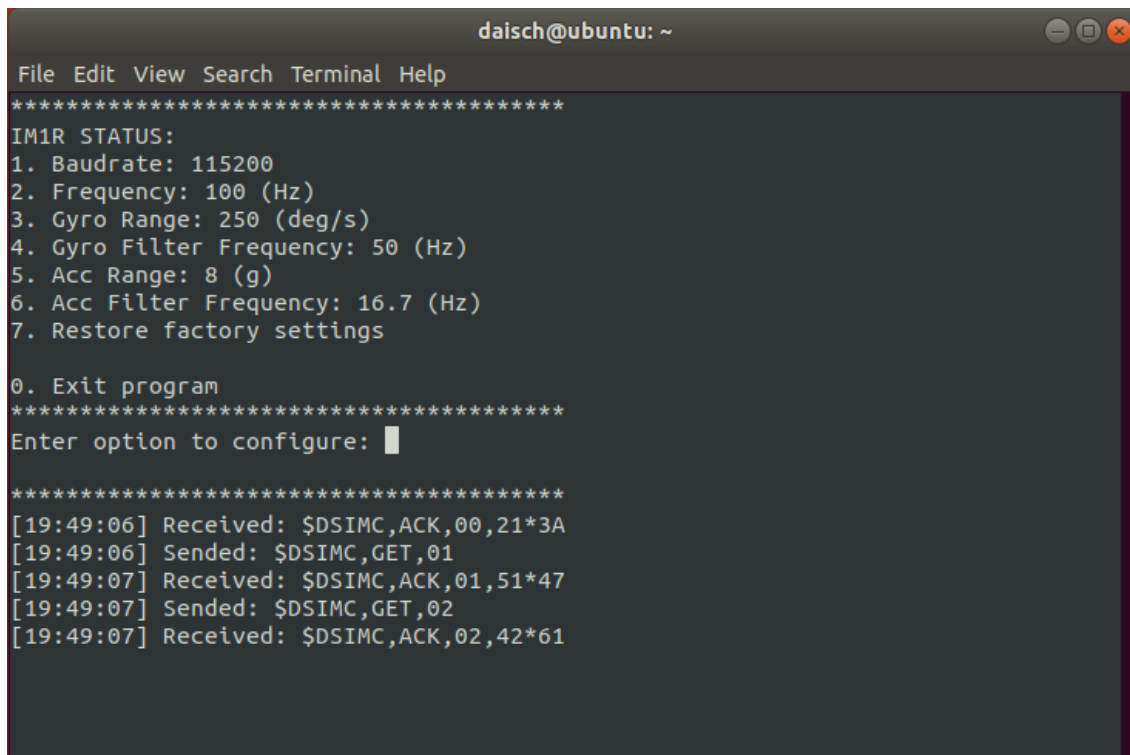
```
sudo chmod 666 /dev/ttyUSB0
```

5. 启动配置工具：

```
roslaunch im1r_ros_driver daisch_im1r_config.py /dev/ttyUSB0
```

◦ /dev/ttyUSB0 是当前IM1R连接的串口

6. 启动成功后，主界面会显示以下内容：

A terminal window titled 'daisch@ubuntu: ~' showing the output of the IM1R configuration tool. The terminal displays a menu of options for configuring the IM1R device, including Baudrate, Frequency, Gyro Range, Gyro Filter Frequency, Acc Range, Acc Filter Frequency, and a 'Restore factory settings' option. The user has entered '0' to exit the program. The terminal also shows a series of received and sent messages from the device, indicating successful communication.

```
daisch@ubuntu: ~
File Edit View Search Terminal Help
*****
IM1R STATUS:
1. Baudrate: 115200
2. Frequency: 100 (Hz)
3. Gyro Range: 250 (deg/s)
4. Gyro Filter Frequency: 50 (Hz)
5. Acc Range: 8 (g)
6. Acc Filter Frequency: 16.7 (Hz)
7. Restore factory settings

0. Exit program
*****
Enter option to configure: 0

*****
[19:49:06] Received: $DSIMC,ACK,00,21*3A
[19:49:06] Sended: $DSIMC,GET,01
[19:49:07] Received: $DSIMC,ACK,01,51*47
[19:49:07] Sended: $DSIMC,GET,02
[19:49:07] Received: $DSIMC,ACK,02,42*61
```

7. 如果串口连接失败，你会看到下图中的信息，请检查设备是否连接正确，并重新设置串口权限。

```
daisch@ubuntu: ~  
File Edit View Search Terminal Help  
Try opening the SerialPort: /dev/ttyUSB0  
[Errno 5] could not open port /dev/ttyUSB0: [Errno 5] Input/output error: '/dev/  
ttyUSB0'  
Try reconnecting the USB device and running the program again  
Press any key to exit.  
█
```

更改配置项

以下是以更改输出频率为例的步骤：

1. 在主界面输入 2 并按回车，进入子菜单：

```
daisch@ubuntu: ~  
File Edit View Search Terminal Help  
*****  
FREQUENCY [Hz]  
1. 10  
2. 20  
3. 50  
4. 100  
5. 250  
6. 500  
7. 1000  
  
0. Return to Main Menu  
*****  
Select configuration option: █  
  
*****  
[19:49:06] Received: $DSIMC,ACK,00,21*3A  
[19:49:06] Sended: $DSIMC,GET,01  
[19:49:07] Received: $DSIMC,ACK,01,51*47  
[19:49:07] Sended: $DSIMC,GET,02  
[19:49:07] Received: $DSIMC,ACK,02,42*61
```

2. 在子菜单中选择所需的输出频率。配置成功后，工具会自动返回主菜单。例如，设置为 50Hz 输出频率后，你可以在主菜单中看到设置已成功更改：

```
daisch@ubuntu: ~
File Edit View Search Terminal Help
*****
IM1R STATUS:
1. Baudrate: 115200
2. Frequency: 50 (Hz)
3. Gyro Range: 250 (deg/s)
4. Gyro Filter Frequency: 50 (Hz)
5. Acc Range: 8 (g)
6. Acc Filter Frequency: 16.7 (Hz)
7. Restore factory settings

0. Exit program
*****
Enter option to configure: █

*****
[19:59:04] Received: $DSIMC,ACK,00,19*3D
[19:59:04] Sended: $DSIMC,GET,01
[19:59:05] Received: $DSIMC,ACK,01,51*47
[19:59:05] Sended: $DSIMC,GET,02
[19:59:05] Received: $DSIMC,ACK,02,42*61
```

注意事项

- 输出频率应与波特率匹配，设置不匹配的波特率和输出频率会导致接收到错误的报文。

波特率	最大输出频率
115200	100Hz
230400	250Hz
460800	500Hz
921600	1000Hz

常见问题解答

问题1：找不到脚本

症状：运行 `roslaunch im1r_ros_driver subscriber_example.py` 时出现以下错误：

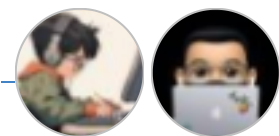
```
[roslaunch] Couldn't find executable named subscriber_example.py below
/home/daisch/catkin_ws/src/im1r_ros_driver
[roslaunch] Found the following, but they're either not files,
[roslaunch] or not executable:
[roslaunch]
/home/daisch/catkin_ws/src/im1r_ros_driver/scripts/subscriber_example.py
```

解决方案：授予脚本可执行权限：

```
chmod +x /home/daisch/catkin_ws/src/im1r_ros_driver/scripts/subscriber_example.py
```

如果您有任何其他问题或需要进一步的帮助，请随时提出。

贡献



许可证

[BSD-3-Clause](#)