**深 圳 大 学 实 验 报 告**

**课程名称：­ AI边缘设计系统**

**实验项目名称： ROS2节点话题通信与相机IMU可视化**

**学院： 电子与信息工程**

**专业： 电子信息工程**

**指导教师： 蒙山**

**报告人： 陈应权 学号： 2022280297 班级： 06**

**实验时间： 2024年10月25日**

**实验报告提交时间： 2024年11月1日**

**教务部制**

|  |
| --- |
| 一、实验准备  1. 地平线开发板安装ros2环境；[参考链接](http://originbot.org/guide/image_install/) 2. 搭建开发环境，使用vscode通过ssh远程连接到开发板；[搭建开发环境](http://originbot.org/manual/ide_setup/) 3. 在安装好ros2和搭建好开发环境的基础上，进行下面的实验； |
| 二、基本概念2.1 节点 ROS2中每一个节点也是只负责一个单独的模块化的功能（比如一个节点负责控制车轮转动，一个节点负责从激光雷达获取数据、一个节点负责处理激光雷达的数据、一个节点负责定位等等）    上面举了一个激光雷达的例子，一个节点负责获取激光雷达的扫描数据，一个节点负责处理激光雷达数据，比如去除噪点。那节点与节点之间就必须要通信了，ROS2中一共有如下四种通信方式，本次实验使用话题topic进行通信。   * 话题-topics * 服务-services * 动作-Action * 参数-parameters  2.2 话题 话题是ROS2中最常用的通信方式之一，话题通信采取的是订阅发布模型。一个节点发布数据到某个话题上，另外一个节点就可以通过订阅话题拿到数据。    除了上述这种一个节点发布，一个节点接受的形式外，ROS2话题通信其实还可以是1对n,n对1,n对n的。  **1对n**    **n对1（同一个话题可以有多个发布者）**    **n对n**    **还有一种就是ROS2节点可以订阅本身发布的话题**   [2.3 工作空间](https://fishros.com/d2lros2/" \l "/humble/chapt2/get_started/2.ROS2%E5%8A%9F%E8%83%BD%E5%8C%85%E4%B8%8E%E5%B7%A5%E4%BD%9C%E7%A9%BA%E9%97%B4?id=_1-%e5%b7%a5%e4%bd%9c%e7%a9%ba%e9%97%b4) 工作空间是包含若干个功能包的目录，可以把工作空间理解成一个文件夹。这个文件夹包含下有src文件夹，src文件夹中存放功能包。 [2.4 功能包](https://fishros.com/d2lros2/" \l "/humble/chapt2/get_started/2.ROS2%E5%8A%9F%E8%83%BD%E5%8C%85%E4%B8%8E%E5%B7%A5%E4%BD%9C%E7%A9%BA%E9%97%B4?id=_2%e5%8a%9f%e8%83%bd%e5%8c%85%e6%98%af%e4%bb%80%e4%b9%88) 功能包可以理解为存放节点的地方，ROS2中功能包根据编译方式的不同分为三种类型。   * ament\_python，适用于python程序 * cmake，适用于C++ * ament\_cmake，适用于C++程序,是cmake的增强版 |
| 实验过程1.创建工作空间 在userdata目录下，创建工作空间my\_dev\_ws，在my\_dev\_ws文件夹下，新建src文件夹用于存放功能包。  $ cd /userdata  $ mkdir my\_dev\_ws  $ cd my\_dev\_ws  $ mkdir src  $ colcon build  编译完成之后，此时在my\_dev\_ws文件夹下会生产另外的三个文件夹，build（编译过程中的中间文件）、install（编译完成的文件）、log（日志文件）。此时没有功能包，所以这三个文件夹应该是空的。  1699360723728(1) 2.创建功能包 切换到工作空间my\_dev\_ws下的src目录中，然后使用ros2 pkg 命令创建功能包topic\_example（功能包名称可自己取）。  $ cd /userdata/my\_dev\_ws/src  $ ros2 pkg create topic\_example --build-type ament\_python --dependencies rclpy  再在功能包topic\_example中的topic\_example文件夹中创建两个文件，一个发布者，一个订阅者，创建完的目录结构如下所示。   3.编写节点 功能包节点的编写，可以使用C++，也可以使用python，以python为例： 3.1发布者节点 创建话题发布者一般包括：   * 创建发布者对象（消息类型、话题名、队列长度） * 创建一个定时器（单位为秒的周期，定时执行的回调函数） * 创建定时器周期执行的回调函数 * 填充消息对象中的消息数据 * 发布话题消息 * 创建ROS2节点对象并进行初始化 * 循环等待ROS2退出    3.2订阅者节点 创建话题订阅者一般包括：   * 创建订阅者对象（消息类型、话题名、订阅者回调函数、队列长度） * 创建回调函数，执行收到话题消息后对数据的处理 * 创建ROS2节点对象并进行初始化 * 循环等待ROS2退出    4、配置编译 修改功能包topic\_example中的配置文件，添加节点的入口函数，如下图所示，等号后面依次是功能包名称、节点文件名称以及入口函数名称。    修改完配置文件之后，进行编译。编译要在工作空间目录下进行，使用colcon进行编译，编译完成后要刷新环境变量。   5、运行 打开一个终端，切换到工作空间目录下，刷新环境变量后，使用ros2 run 命令执行消息发布节点；  打开另外一个终端，切换到工作空间目录下，刷新环境变量后，使用ros2 run 命令执行消息订阅节点； 6.WSL2桥接模式安装hyper-V 新建文件夹hyper-V.bat文件，复制下面内容到文件中：  @echo off  pushd "%~dp0"  dir /b %SystemRoot%\servicing\Packages\\*Hyper-V\*.mum >hyper-v.txt  for /f %%i in ('findstr /i . hyper-v.txt 2^>nul') do dism /online /norestart /add-package:"%SystemRoot%\servicing\Packages\%%i"  del hyper-v.txt  Dism /online /enable-feature /featurename:Microsoft-Hyper-V-All /LimitAccess /ALL  Pause  以管理员身份运行该脚本。 启动Windows系统功能 路径：搜索→控制面板→程序和功能→启动和关闭Windows系统功能  在这里插入图片描述 Type-V管理器环境配置 搜索→Hyper-V管理器→虚拟交换机管理器→新建虚拟机交换机→创建虚拟交换机  在这里插入图片描述  在这里插入图片描述 7.DDS安装 cyclonedds在OriginBot的镜像中已经配置好，需要在电脑端wsl中Ubuntu配置，使用如下命令安装：  $ sudo apt install ros-humble-rmw-cyclonedds-cpp 8.ROS图像传输启动ROS相机驱动 SSH连接OriginBot成功后，在X3终端中输入如下指令：  $ export RMW\_IMPLEMENTATION=rmw\_cyclonedds\_cpp  $ export CYCLONEDDS\_URI='<CycloneDDS><Domain><General><NetworkInterfaceAddress> wlan0</NetworkInterfaceAddress></General></Domain></CycloneDDS>'  $ ros2 launch originbot\_bringup camera.launch.py  2022-09-26_12-45  wlan0为网卡名称，可通过ifconfig查看 查看相机话题 运行成功后，可在机器人端或同网络下的PC端即wsl中Ubuntu使用如下命令，确认图像话题已经正常发布。  $ ros2 topic list  image-20220922172506701  相机话题能成功在wsl中Ubuntu接收，需要Ubuntu与X3处于同一网关192.168.137.X，可通过指令ifconfig来确认。 查看可视化图像 在同一网络中的PC端，同样配置好使用的DDS，再使用rqt\_image\_view工具，就可以看到机器人的实时相机图像了。  $ export RMW\_IMPLEMENTATION=rmw\_cyclonedds\_cpp  $ export CYCLONEDDS\_URI='<CycloneDDS><Domain><General><NetworkInterfaceAddress>wlan0</NetworkInterfaceAddress></General></Domain></CycloneDDS>'  $ ros2 run rqt\_image\_view rqt\_image\_view  wlan0为网卡名称，可通过ifconfig查看  2022-09-26_12-48_1   9.IMU驱动与可视化IMU驱动 SSH连接OriginBot成功后，在终端中输入如下指令，即可启动机器人底盘及IMU：  $ ros2 launch originbot\_bringup originbot.launch.py use\_imu:=true  image-20220822150712051 数据可视化 在同一网络中的PC端，安装Rivz中的IMU插件：  $ sudo apt install ros-${ROS\_DISTRO}-rviz-imu-plugin  然后启动上位机可视化软件：  $ ros2 launch originbot\_viz display\_imu.launch.py  即可看到可视化的IMU信息，此时摇动机器人，Rviz中的坐标系也会跟随运动。  img |

深圳大学学生实验报告用纸

|  |
| --- |
| 实验过程：  实验中的节点通信部分：  1.加入了日期和修改了自己的姓名和学号部分，主要通过datetime函数来获取时间。    2.  桥接模式的设置：  按照上述教程设置好虚拟交换机后，在用户的页面添加.wslconfig文件    在.wslconfig文件中，编辑以下内容    关闭电脑的防火墙，关闭代理，打开cmd，打开wsl，  配置/lib/systemd/network/wsl\_external.network 处文件  **[**Match**]**  Name**=**eth0  **[**Network**]**  Description**=**bridge  DHCP**=**falseAddress**=**192.168.1.110/24  Gateway**=**192.168.1.1  添加/etc/wsl.conf  **[**boot**]**  systemd**=**true  **[**network**]**  generateResolvConf **=** false  重启wsl，即可成功完成桥接模式 |
| 实验结果：  节点通信：      摄像机：  1730398276888  IMU:    1730395714788 |
| 实验结论：  经过本次实验，我们对ROS2（Robot Operating System 2）节点间的话题通信有了深入的理解和实践。以下是实验的主要结论：  ROS2节点通信机制：  我们成功地搭建了ROS2环境，并创建了多个节点，这些节点能够通过话题（topics）进行通信。  实验结果表明，节点间的话题通信是ROS2中实现模块化和分布式控制的关键机制。  通信效率与可靠性：  通过实验，我们评估了不同通信参数下的话题通信效率和可靠性。  结果显示，在适当的参数配置下，ROS2的话题通信既高效又可靠，满足了实时性要求。  错误处理与调试：  实验中，我们遇到了一些通信错误，通过分析和调试，我们学会了如何识别和解决这些问题。  我们发现，正确的错误处理机制对于确保系统稳定性和可靠性至关重要 |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。