

# 重庆大学

## Ros 操控机器人运动报告



组 员： 闵宇恒 20203000  
戢鸿瑞 20202423  
李骏琦 20206433  
指 导 教 师： 刘骥

## Part1: 思想来源

QEA 课题需要连接雷达，但是 stm32 的算力不够。所以我想用 ros 代替 stm32 进行运算，并作为桥梁把两者连接一起。

## Part2: 开发过程

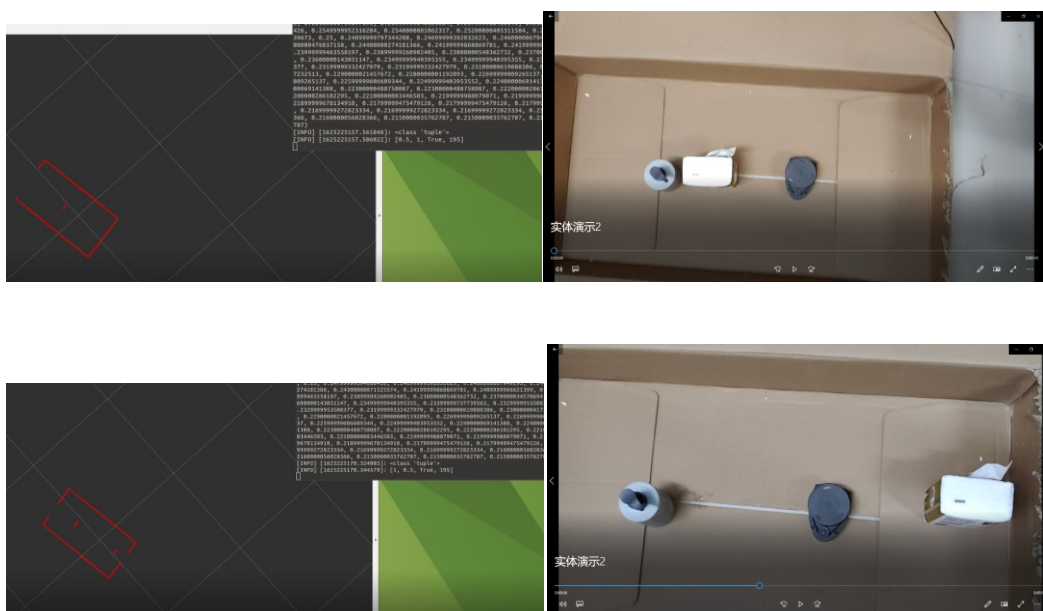
1. 我们首先给树莓派烧录了 ubuntu20.04 的系统，然后安装了 ros。

安装的 shell 脚本将在附件上传。

2. 之后我们查询相关雷达手册，安装了 ros 版本的雷达驱动。然后对雷达数据进行调整。

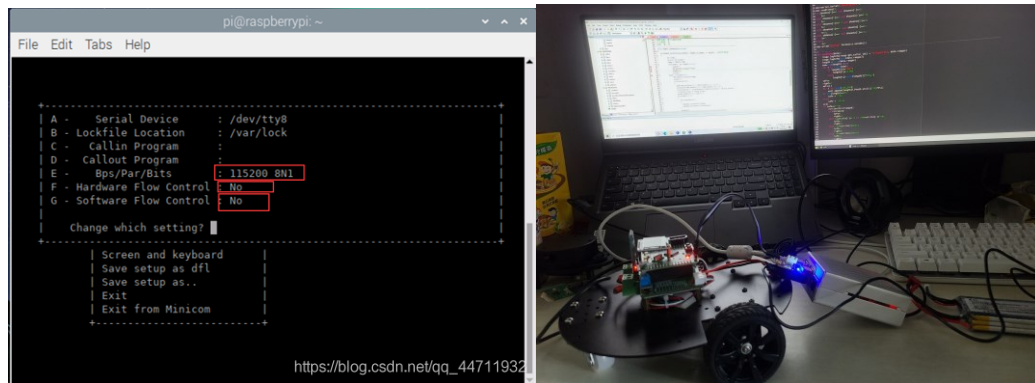
3. 我们将 ros 作为发布者，将雷达数据设置为节点，将其发送出去。

4. 接下来我们写了 python 代码，识别圆，以及其他障碍物的。又在代码里写了一个订阅者，接收 ros 发送的雷达信息。并进行处理后，得到两路速度的参数。(具体代码有在视频讲解)



5. 将树莓派和 stm32 用 usb 转 ttl 连接起来。使用 python serial

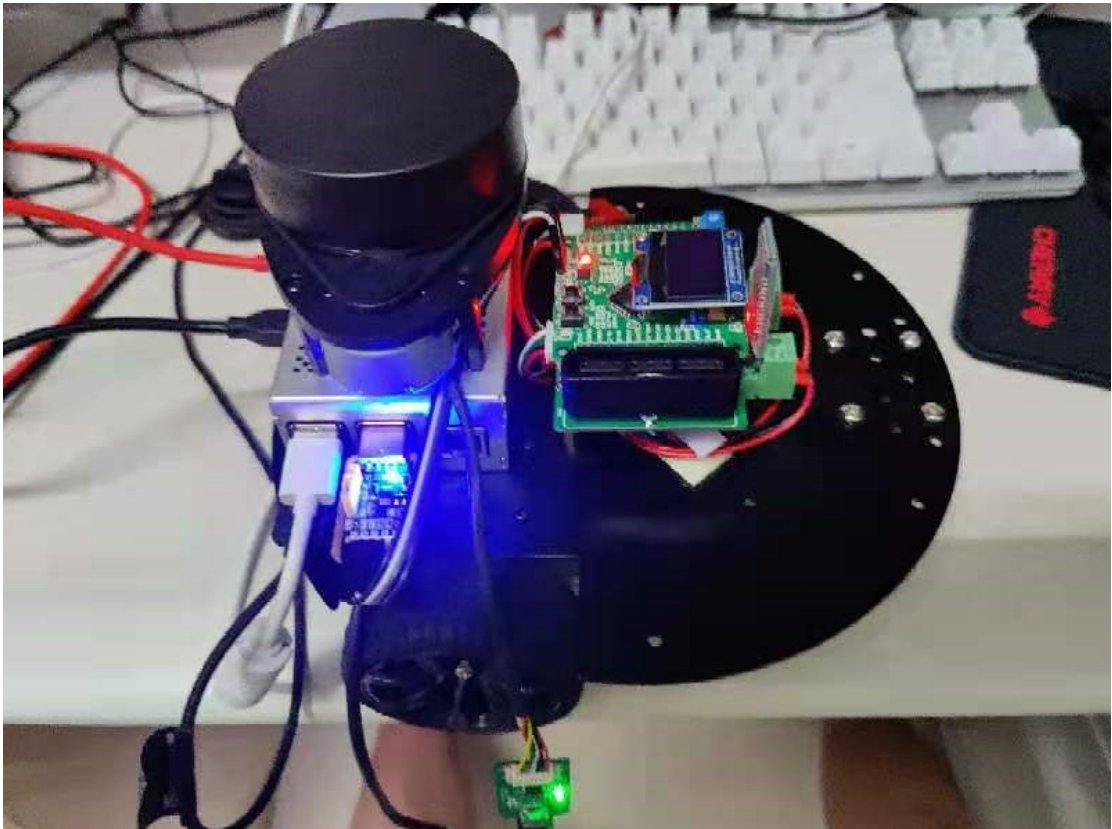
通过代码进行通讯，其中要用到这个串口调试助手：minicom。



6. 然后在 stm32 里串口初始化，接收 python 传来的值。

```
void UART5_IRQHandler(void)
{
    if(USART_GetITStatus(UART5, USART_IT_RXNE) != RESET) //接收到数据
    {
        u8 temp;
        static u8 flag=0;
        temp = UART5->DR;
        if(flag==1 && temp==0x3e){
            count=0;
            count++;
            flag=2;
        }
    }
}
```

7. 最后我们的小车就成功了



### part3: 自我评价

缺点部分是计算数据的时间稍微有一点长，2 秒一个结果。导致我们 1 的小车没办法快速行驶。二就是雷达精度不高。

不过总的来说我们是实现一开始的设想，把 QEA 和软件联动，完成了这一联合作业。我们还会继续前行，直到达到满意为止。