7、二维码指令



7、二维码指令

7.1、实验说明

7.2、实验目标

7.3、实验操作

7.4、实验效果

7.5、实验总结

7.1、实验说明

本实验是属于拓展类实验,需要搭配其他外部设备使用,这里用到了小车底盘和ROS扩展板都不属于 K210开发板套件内容,所以本实验效果仅供参考,如果没有对应的设备是不能直接使用本例程代码。

ROS扩展板需要提前烧录固件: ROS-CAR.hex

由于本次使用的电机电压为8.4V,所以ROS扩展板的电池不可以插入12.6V电池,需要插入8.4V电池。

小车电机连接线如下图所示:

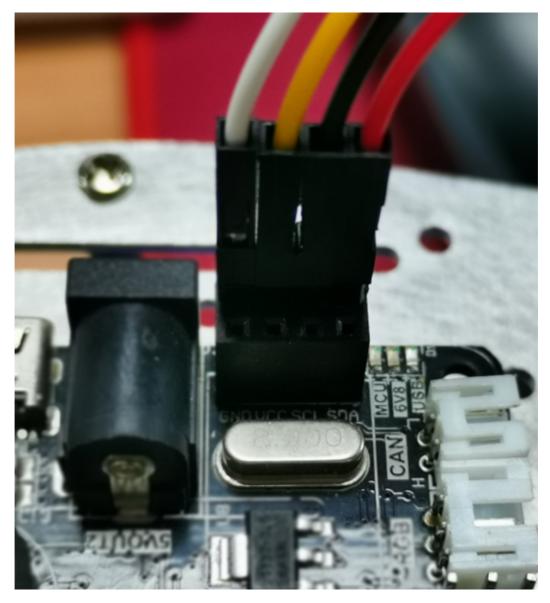
电机Motor 1连接左前轮,电机Motor 2连接左后轮,电机Motor 3连接右前轮,电机Motor 4连接右后轮。



K210开发板与ROS扩展板连接的线序如下图所示:

白色连接GND, 黄色连接VCC, 黑色连接SCL, 红色连接SDA。

这里需注意一下,图示中的标识为I2C线序标识,但是K210使用的是串口通讯,由于烧录的ROS-CAR.hex文件已经把这个接口修改为串口信号,所以实际上ROS扩展板上的接口对应关系是:SCL实际上是TX,SDA实际上RX。



7.2、实验目标

本节课主要学习K210开发板搭配小车底盘做视觉巡线的功能。

本次实验的参考代码路径为: 06-export\qrcode_motion.py

7.3、实验操作

- 1. ROS扩展板烧录固件: ROS-CAR.hex
- 2. 将RGB灯条插入ROS扩展板的RGB灯接口。
- 3. 请将06-export\library目录下的小车驱动库和PID控制库提前下载到内存卡的根目录上。
- 4. 打开CanMV IDE打开qrcode_motion.py代码并下载到K210开发板上。
- 5. 将K210开发板通过4PIN排线连接到ROS扩展板上。
- 6. 将小车放入白色背景中,掰动K210开发板支架到合适的角度,打开小车的开关。

7.4、实验效果

等待系统初始化完成后,小车在没有识别到二维码时不运动,当识别到带有"turn left"字符的二维码时,会执行左转,当识别到带有"turn right"字符的二维码时,会执行右转,"spin left"字符是左旋,"spin right"字符是右旋,"go ahead"字符是向前走,"fall back"字符是向后退。识别到对应字符后运行功能,一定时间后会自动停止,停止后可再次识别二维码。

```
def parse_payload(payload):
global car count
if payload == "turn left":
    print("car turn left")
    car count = 20
    bot.set car motion(0, 0, 3)
elif payload == "turn right":
    print("car turn right")
    car count = 20
    bot.set_car_motion(0, 0, -3)
if payload == "spin left":
    print("car spin left")
    car_count = 40
    bot.set_car_motion(0, 0, 3)
elif payload == "spin right":
     print("car spin right")
    car count = 40
    bot.set car motion(0, 0, -3)
elif payload == "go ahead":
    print("car go ahead")
    car count = 20
    bot.set_car_motion(0.5, 0, 0)
elif payload == "fall back":
    print("car fall back")
    car count = 20
     bot.set_car_motion(-0.5, 0, 0)
```

7.5、实验总结

小车识别二维码执行指令功能是基于二维码识别功能而来,识别二维码后立即执行对应的动作,如果识别到的二维码字符不是预设的内容,则不做任何运动。