

《机器人编程实践》学生实验报告

学院 数计学院 专业 计算机科学与技术 班级 计科3班
姓名 周吉瑞 学号 20190521340 日期 2021/06/03

课程名称： 机器人编程实践
实验名称： 实验九 定点停车
指导老师： 孙建勇

目录

一、实验记录及总结

1. 实践任务（1）
2. 方案实践与记录（1）
3. 实践任务（2）
4. 方案实践与记录（2）

二、拓展任务

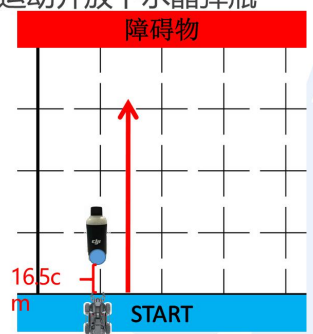
1. 实践任务
2. 方案实践与记录

一、实验记录及总结

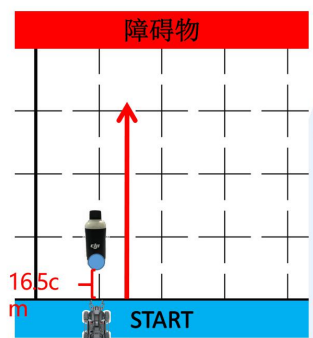
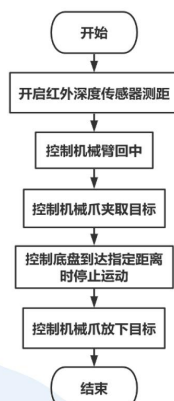
1、实践任务（1）

- ◆ 夹取水晶弹瓶，检测到离障碍物的距离小于30厘米时，控制底盘停止运动并放下水晶弹瓶

- 1、开启红外深度传感器
- 2、控制机械臂回中
(机械爪初始状态为闭合)
- 3、控制机械爪夹取水晶弹瓶
- 4、检测到指定距离时底盘停止运动
- 5、放下水晶弹瓶



流程图：



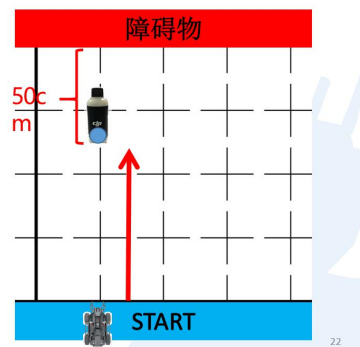
2、方案实践与记录（1）



3、实践任务（2）

◆ 检测距离障碍物小于50厘米时，向正前方自动夹取水晶弹瓶

- 1、开启红外深度传感器
- 2、控制机械臂回中
(机械爪初始状态为闭合)
- 3、检测到指定距离时底盘停止运动
- 4、夹取水晶弹瓶



4、方案实践与记录（2）



二、拓展任务

1、实践任务

◆ 利用TOF实现类似扫地机器人的效果

扫地机器人迭代与进化：产品从随机碰撞到自主导航，感知模块的进步是关键：感知模块通过对周围环境的感知扫描，实现地图建模、定位和导航，依赖于各种传感器、陀螺仪等。

路径规划步骤：

- 1) 使EP Robot能从初始位置运动到目标位置；
- 2) 编写程序使EP Robot能绕开障碍物，并且经过某些必须经过的点完成相应的作业任务。



2、方案实践与记录

【理论分析】

如何才能从一个点移动到另一个点呢？

设：起点为 (x, y) 终点为 (a, b)

那么从起点移动到终点的步骤应该为：

<1>、x 轴方向上移动 $a-x$ 个长度单位 ($a-x$ 为负数时为向左移动，否则为向右移动)

<2>、y 轴方向上移动 $b-y$ 个长度单位 ($b-y$ 为负数时为向下移动，否则为向上移动)

如何实现自动避开障碍？

全程开启红外测距仪，当障碍范围大于所需移动范围时，才移动，否则跳过本次移动。

【伪代码】

```
// 伪代码 (C 语言风格)

main()
{
    openDiastimeter(1);
    // 开启 1 号红外深度传感器测距

    setSpeed(0.3);
    // 设置底盘平移速率

    double x = 0, y = 0;           // 坐标点
    double temp_x = 0, temp_y = 0; // 零时保存坐标点
    double a = 0, b = 0;           // 上一个坐标点

    // 每次输入一个坐标
    while (scanf("%lf%lf", &x, &y)) {
```

```

temp_x = x;

// 用零时变量将坐标初始值占时保存

temp_y = y;

// 用零时变量将坐标初始值占时保存


x = x - a;

// 去往该点的 x 轴移动单位=此点 x 坐标-上一个点 x 坐标

y = y - b;

// 去往该点的 y 轴移动单位=此点 y 坐标-上一个点 y 坐标


// 控制底盘在 x 轴上的移动

if (x >= 0) {

    setCourseAngle(90);

    if (getDiastimeter() >= x * 100) {

        move(90, abs(x));

        // 如果 x 为正, 那么底盘向 90 度方向移动绝对值 x 个长度单位

    }

}

else {

    setCourseAngle(90);

    if (getDiastimeter() >= x * 100) {

        move(-90, abs(x));

        // 如果 x 为负, 那么底盘向-90 度方向移动绝对值 x 个长度单位

    }

}


// 控制底盘在 y 轴上的移动

if (y >= 0) {

    setCourseAngle(90);

    if (getDiastimeter() >= y * 100) {

        move(0, abs(y));

        // 如果 y 为正, 那么底盘向 0 度方向移动绝对值 y 个长度单位

    }

} else {

    setCourseAngle(90);

    if (getDiastimeter() >= y * 100) {

        move(-180, abs(y));

        // 如果 y 为负, 那么底盘向-180 度方向移动绝对值 y 个长度单位

    }

}


a = temp_x;

// 保存该点的 x 值

b = temp_y;

```

```
// 保存该点的 y 值
wait(3);

// 等待 3 秒
}
```

【主函数模块】

The main function block is a vertical sequence of modules. It starts with a '开始运行' (Start Running) module, followed by '设置整机运动' (Set Whole Machine Motion) set to '自由模式' (Free Mode). Then, it activates the '1' channel of the '号红外深度传感器测距' (Infrared Depth Sensor Distance Measurement) module. The sequence then proceeds through four 'next' blocks, each containing two '将' (Set) modules to update x and y coordinates: (0, 0), (0.6, 0.3), (0.6, 0.6), and (1.2, 0.6).

【next 函数模块】

The 'next' function module is a vertical sequence of four modules: 'temp1', 'coordinate', 'move', and 'temp2'. It concludes with a '等待' (Wait) module set to 3 seconds.

【coordinate 函数模块】

The 'coordinate' function module is a vertical sequence of two '将' (Set) modules. The first sets 'x' to 'x - a', and the second sets 'y' to 'y - b'.

【move 函数模块】



【temp1 函数模块】



【temp2 函数模块】



【avoid 函数模块】



【avoid 函数模块】

