目标检测及抓取型机器人

**开发计划**

**SDP203**

**v1.0**

分工说明

|  |  |  |
| --- | --- | --- |
| 小组名称 | Whitegivers | |
| 学号 | 姓名 | 本文档中主要承担的工作内容 |
| 17373344 | 杨华晟 | 主编，负责第1234小节的编写 |
| 77086002 | 萨扎尔 | 负责第三小节编写 |
| 17373342 | 邓力友 | 负责6节的编写 |
| 17373332 | 刘冬 | 负责第5节编写 |

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| 1.0.0 | 2020.3.9 | 杨华晟 | 邓力友 | 初步完成开发计划制定，对于模块细节划分和分工有待之后进行调整。数据库开发需求暂时没有。 |
| 2.0.0 | 2020.5.8 | 杨华晟 | 邓力友 | 对使用gazebo进行机器人仿真程序搭建作出计划，并对原有计划进行修改 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# **范围**

## **项目概述**

本项目的嵌入式系统为启智ROS机器人。ROS 是一个构建在 Ubuntu 上的机器人开发系统，由一系列的软件包和通讯架构组成。

Ø 主要功能：通过嵌入式编程手段实现在启智ROS机器人上进行目标物体识别和目标物体抓取，同时机器人具有障碍识别和自动路径规划功能。

Ø 非功能性需求：在主要功能实现后将会考虑实现以下额外功能：语音交互。

Ø 应用场景：家庭中替人类拿取或放置物品，可用来帮助身体残疾或不便人士。工厂流水线中机械臂进行自动工件组装和物体搬运。公共场所可作为垃圾机器人自动识别并打扫垃圾。

## **文档概述**

本文档用于清晰划分组内个成员的分工与合作，设定项目进行计划和任务分工，从而指导项目有序进行。主要内容有项目任务概要，风险管理，过程模型，资源计划，进度计划几个部分。

## **术语和缩略词**

1. ROS : 机器人操作系统（Robot Operating System）

## **引用文档**

1. 启智ROS机器人开发手册V1.1.0

# **项目任务概要**

## **工作内容**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 任务编号 | 一级任务 | 二级任务 | 负责人 | 参与人员 |
| T1 | 开发计划报告 | 无 | 杨华晟 | 全体成员 |
| T2 | 需求分析与文档 | 无 | 刘冬 | 全体成员 |
| T3 | 软件需求分析设计 | 无 | 邓力友 | 全体成员 |
| T4 | 软件开发 | 机器人移动模块编写 | 杨华晟，邓力友 | 杨华晟 |
| 地图建立模块编写 | 邓力友 |
| 路径规划模块编写 | 邓力友 |
| 图形用户界面编写 | 刘冬萨扎尔 |
| 模块集成 | 全体成员 |
| T5 | 后端测试 | 机器人底层模块测试 | 邓力友 | 杨华晟 |
| 前端测试 | 用户界面测试 | 刘冬 | 杨华晟 |
| T6 | 测试文档编写 | 无 | 杨华晟 | 邓力友 |

## **主要人员**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 主要人员 | 角色 | 工作时间 | 工作经验 | 技术水平 |
| 杨华晟 | 项目经理，程序开发，文档评审 | 工作时长两年半 | 代码量约为2w行，在大三上的编译理论课程设计中独立完成类C语法的编译器的设计与实现。嵌入式系统开发经验较少。 | 对C，C++的使用较为熟悉。掌握面向对象的编程思想和技术，熟悉基础的算法和数据结构。  对python的使用相对熟悉 |
| 邓力友 | 程序开发，系统架构师 | 两年半 | 无嵌入式系统开发经验，仅独立完成过一个类C编译器、一个小型cpu等 | 掌握面向对象的编程思想，对C、java、python都有所了解 |
| 萨扎尔 | 程序开发，测试人员 | 两年半 | 在学ColimbiaX的Robotics课程，有两个月的java实习经验 | 对C、C++、java、python、swift的使用较为熟悉。熟悉基础数据结构、面向对象 |
| 刘冬 | 程序开发，测试人员 | 两年半 | 有过短暂java实习经验，无嵌入式开发经验。校内课程完成 | 对C、java较为熟悉。掌握面对对象思想方法和数据结构。 |

## **产品**

### **程序、数据或设备**

产品最终成果为目标检测及抓取机器人的源码以及由源代码编译生成的可执行目标程序。

项目文件夹为robot

项目源代码：robot/src

可执行程序：robot/activate

安装程序：暂无

数据文件：bot/data

### **文档**

1. 开发计划文档
2. 软件需求规格说明书
3. 软件设计说明
4. 软件测试说明

## **运行与开发环境**

### **运行环境**

Ø 硬件环境：启智ROS机器人

Ø 软件环境：Ubuntu16.04

### **开发环境**

Ø 推荐硬件环境：

处理器：2.3GHz Intel Core i5

内存：8GB DDR3

硬盘：100G（至少）

网络：使用 TCP/IP协议的局域网

Ø 软件环境：

操作系统：Ubuntu16.04

Gazebo7/8仿真器

ROS Kinetic

## **项目期限**

起始时间：2020年3月6日

结束时间：2020年6月19日

# **风险管理**

|  |  |  |  |
| --- | --- | --- | --- |
| 风险系数 | 风险项名称 | 风险描述 | 风险管理策略 |
| 5 | 知识与技术储备不足 | 本次项目的嵌入式机器人开发所涉及的知识较多，给开发人员带来一定困难 | 在项目开发前期充分阅读指导手册，并编写demo熟悉编程环境以便后续开发。额外花费时间进行学习。 |
| 3 | 交流沟通存在困难 | 由于疫情影响项目组在前期只能采取线上沟通，带来一定的沟通低效和不及时的问题 | 每周微信进行线上会议，项目经理及时联系组员，开发人员及时汇报工作情况以及所遇到的问题 |
| 3 | 测试不便导致bug无法及时被发现和解决 | 由于机器人远在校区，编码无法及时得到反馈结果 | 合理进行需求分析和模块划分，从而使bug可定位化 |
| 2 | 开发环境不够熟悉 | 在linux系统中开发的经验不足，可能导致项目开发速度减缓 | 额外花费时间学习 |
| 2 | 不可抗力干扰 | 项目开发途中将收到其他课程以及种种ddl的摧残导致项目时间缩水 | 为避免这种情况必须提前进行工作，在前期加大力度 |

# **过程模型**

本项目采取瀑布模型。主要阶段如下：

Ø 需求分析：对项目进行多层次多方面的需求分析，包括且不局限于以下方面：业务需求，数据需求，功能需求，用户界面需求等。

Ø 软件系统设计：包括数据库设计，软件体系结构设计，接口设计。将机器人系统划分成若干个模块。

Ø 程序设计：对系统中的每个关键模块，给出详细的设计方案。

Ø 编码：按照模块有计划的完成编码。

Ø 模块集成：将模块组装从而完成最终的功能。

Ø 系统测试：测试阶段包括测试用例设计，测试进行，测试结果分析，测试结果反馈以及Debugging。

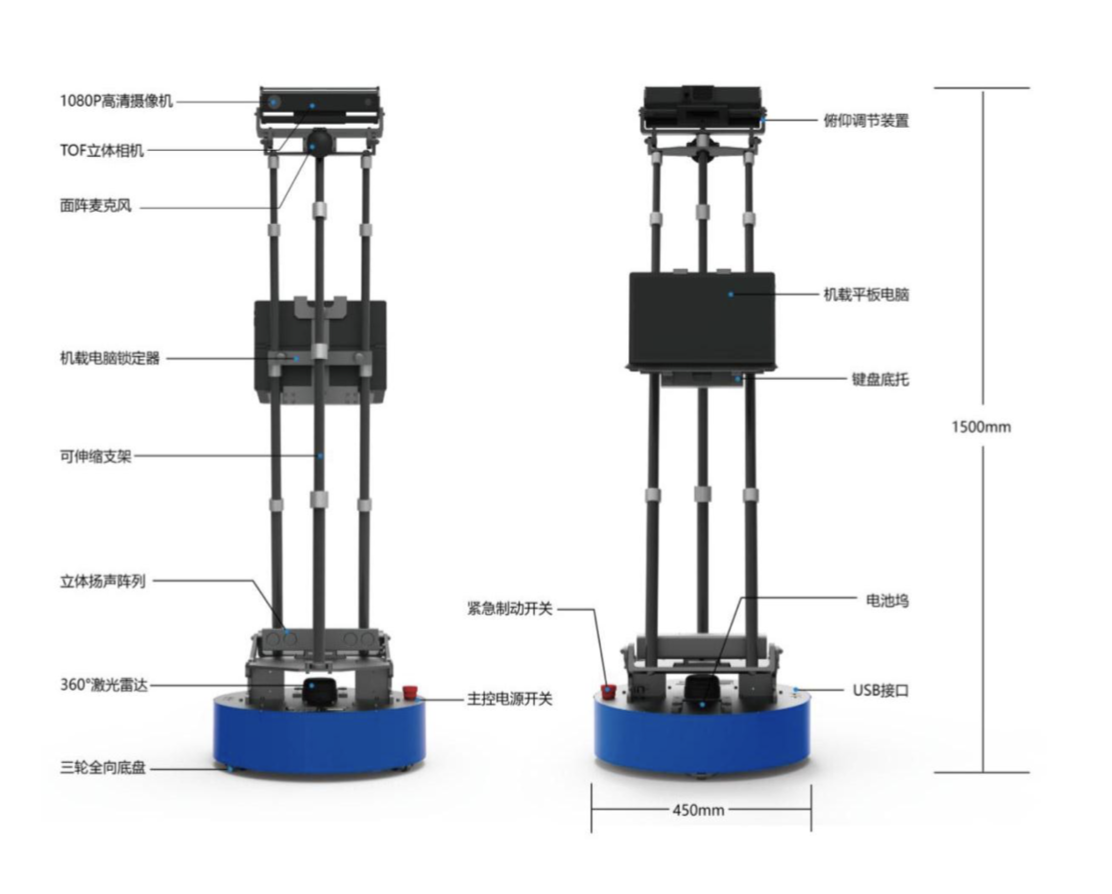
# **资源计划**

Ø 硬件：

(1). 启智ROS机器人一个，并包括：

|  |  |  |
| --- | --- | --- |
| 名称 | 数量 | 参数 |
| 激光雷达 | 1 | 360°无死角、最大距离8米 |
| 视觉传感器 | 1 | Kinect2 |
| 电池 | 1 | 24V3.5AH锂离子动力电池 |
| 轮子 | 3 | 全向轮 |
| 伺服电机模块 | 3 | 20W伺服电机、内置驱动 |

外观视图如下：



(2). 个人电脑4台

(3). 使用TCP/IP协议的局域网

Ø 软件：

操作系统 : ubuntu16.04

开发工具 : ROS(Kinetic), VS Code

# **进度计划**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 任务名称 | 工期 | 开始时间 | 完成时间 | 前置任务 |
| 环境搭建 | 5 个工作日 | 2020年3月9日 | 2020年3月13日 |  |
| 运动控制 | 7 个工作日 | 2020年3月14日 | 2020年3月20日 | 1 |
| 机械臂控制和测试 | 14 个工作日 | 2020年3月14日 | 2020年3月27日 | 1 |
| 需求分析文档 | 22 个工作日 | 2020年3月9日 | 2020年3月30日 |  |
| SLAM建图 | 14 个工作日 | 2020年3月28日 | 2020年4月10日 | 2,3 |
| 编写路径规划和导航程序 | 14 个工作日 | 2020年4月7日 | 2020年4月20日 | 2,3 |
| 软件需求分析文档 | 21 个工作日 | 2020年3月31日 | 2020年4月20日 | 4 |
| 设计图形用户界面 | 14 个工作日 | 2020年4月21日 | 2020年5月4日 |  |
| 前后端对接 | 14 个工作日 | 2020年5月1日 | 2020年5月14日 |  |
| 系统编码完成 | 63 个工作日 | 2020年3月14日 | 2020年5月15日 |  |
| 系统测试和文档编写 | 21 个工作日 | 2020年5月16日 | 2020年6月5日 | 10 |
| 课程总结 | 14 个工作日 | 2020年6月6日 | 2020年6月19日 |  |

甘特图如下：

