

<<<<<< HEAD

# 项目简介

## 规则

=====

# Assembly\_robot

---

## 项目简介

## 规则

### 1. 传送带控制

- 取料传送带通过 220V/60W 电机控制运行，电机配有可调速控制盒，运行速度在 1-6 米/分钟之间调节，可以通过控制盒旋钮进行手动调节，比赛开始前由参赛队员调试设定，传送带运行时速度不得低于 1 米/分钟，比赛过程中不得再调节修改。
- 传送带上配有 12V 继电器模块，比赛中机器人可以通过 12V 继电器模块控制取料传送带启动或停止。

### 1. 提供标准 220V 电源接口

- 工作电源可以采用电池供电，也可以使用现场提供 220V 电源。

### 1. 待装配体旋转平台控制

- 装配机器人比赛项目场地中的待装配体旋转平台通过 24V 步进电机控制旋转，步进电机驱动器采用雷赛 DM556 型，工作电压 24V。比赛时驱动器的电机端接线参赛队不得进行改动，控制端由各参赛队接入 3.3V 或 5V 电压的 PUL、DIR、EN 三种控制信号。
- 比赛场地中旋转圆台供电可以是外接 220V 电源供电或者电池供电。
- 步进电机为普通 57 型，配雷赛 DM556 型驱动器，皮带传动机构小同步轮齿数 30 齿，大同步轮齿数 150 齿，两个同步轮中心距 190mm。同步带宽 15mm。
- 比赛的预赛阶段，待装配体平台受各自上场机器人控制，待装配体平台可以保持不动也可以旋转，机器人通过步进电机驱动器控制旋转平台的启停和旋转速度。控制端接线以及驱动器 SW1-SW8 配置由参赛队员在调试阶段自行设置确认正确。
- 比赛的决赛阶段，待装配体平台不受上场机器人控制，待装配体平台以每隔 10 秒转动45 度的节拍进行不断旋转工作。

### 1. 自制要求

1. 机器人整体结构自己设计制作，能提供全套总装图和零件加工图。
2. 机器人控制板自己设计制作，能提供完整详细原理图和 PCB 图。
3. 机器人中除操作系统之外的控制程序需参赛队员自己编写，官方库函数可以使用，但必须能展示到最底层代码并参赛队员能够演示讲解。

<<<<<< HEAD

## 结构划分

=====

## 结构划分

- 机械结构
  - 直线模组
  - 42步进电机
  - 步进电机驱动器
  - 谐波减速器
  - 气动夹爪
  - 其他零件
  - DOBOT模型
- 电气结构
  - 电源降压电路 ( DCDC )
  - 电机驱动电路
  - 主控模块
  - 通信模块
  - 视觉传感器
  - 树莓派
  - 继电器控制
- 软件设计
  - 形状颜色识别
  - 串口数据通信
  - 姿态解算

## 项目流程图

1. 软件部分
  1. 树莓派相机标定
  2. 颜色识别
  3. 串口通信
  4. 信息包
2. 外设资源分配
3. 原理图及pcb
4. 机械结构
5. 调试

## 相机识别步骤

1. 相机标定
  - 相机标定的合格指标一共有四个，分别是x轴方向、y轴方向、远近以及旋转角度。
2. 相机和机械臂进行手眼标定
3. 读取摄像头信息
4. 识别出目标物体并得出位姿
5. 物体在相机中的二维坐标转化成相对于相机光心的三维坐标
6. 再转换成相对于机械臂底座的世界坐标
7. 计算机械臂末端执行器抓取物体时的位姿

8. 通过已知的目标位置和末端执行器的位姿信息来计算机械臂上每个电机应转动的角度
9. 根据有无障碍物、路径距离要求、时间要求规划出合适的路径
10. 驱动机械臂各个关节一步一步到达合适的位置
11. 末端执行器抓取物体
12. 投放到指定位置

## 树莓派使用

1. 连接type-c电源线（系统都已经烧录好，无需再烧录）
2. ssh连接
  1. 更新ssh固件/下载finalshell
  2. 连接无线网络（可以手机或电脑开启热点）
  3. ssh远程连接命令（在cmd或者finalshell中设置好用户名和IP地址

```
ssh xue@192.168.137.109//ip地址有可能会变化，具体通过在windows中使用ipconfig命令确定
password: 123456
```

4. 使用VNC连接远程桌面
5. 连接usb摄像头

```
sudo motion //启动摄像头
```

4. 浏览器输入http://192.168.137.109:8080
5. 选择文件路径

```
cd /home/Assembly_robot
```

6. 写入Python文件使用

```
vim + doc_name
```

## 树莓派通信

1. 确定帧头和帧尾（开头两位为帧头，中间最多5位为数据帧，最后一位为帧尾）
2. 树莓派已经打开串口通信，需要使用时用"/dev/AttyS0"
3. stm32上打开USART1（具体接线见引脚标号文档）

## 外设资源分配

1. 共控制四个电机（1个控制底座旋转，1个控制底座平移，两个控制手臂上下移动）
  1. 需要用到几个定时器（暂未确定）
  2. 控制方向引脚未确定

2. 使用USART1串口与树莓派连接

## 机械臂逆运动学求解

1. 确定目标位置
2. 求解各电机转动的角度[bilibili视频](#)

## 颜色识别

1. 转换为HSL格式的图像，选中白色为检测的颜色
2.  $H=[1,179], L=[0,255], S=[0,255]$
- 3.

## 目前主要需要攻克的问题

1. 定时器控制4个电机输出不同pwm接线和方案（各引脚的分配）
2. 材料BOM表
3. 直线模组的主要零件
4. 树莓派与stm32通信问题
5. 机械臂外观设计[参照这个视频](#)
6. 树莓派摄像头生成图片后对图片进行实施处理
7. 树莓派开机自动执行程序 mystart.service文件
8. 相机标定确定需要传回的参数

## 工作流程

1. 相机识别当前位置，回归初始位置
2. 相机标定目标物体的位置
3. 接收 $x, y, h, r$ 将其中心点对应于机械爪中心
4. 通过位姿分析转换为电机1·2·3·4分别的转动角度（添加pid调节使其快速响应）
5. 添加卡尔曼滤波来减小误差（后期进行）

## 变量定义

- 物体x轴坐标位置 $x$
- 物体y轴坐标位置 $y$
- 物体与相机距离 $h$
- 相机与物体的偏转角度 $r$
- 底部平移电机的转动角度 $j1$
- 底部旋转电机的转动角度 $j2$
- 大臂电机的转动角度 $j3$
- 小臂电机的转动角度 $j4$

后续正在补充ing.... <<<<<< HEAD

=====