控制策略

控制整车的主控是两块STM32F407IGT6芯片（焊在两块自主设计的PCB上），和一块树莓派3B+。

**控制板1，负责总逻辑控制：**

1. 连接其他控制板，进行通信
   1. 获取控制板2的编码器和陀螺仪信息，通过串口与控制板2进行传输.
   2. 与树莓派3B+通信，获取树莓派摄像头回传的信息
2. 连接外设
   1. 四个CCD摄像头传感器，负责识别地图上的特征点，如黑线，黑线交叉点，确定自己的位置。
   2. 四个步进电机控制器，通过脉冲数控制步进电机旋转。底盘采用全向瑞典轮，用雅各比矩阵算法，解算出小车的全向运动。
   3. 2个CAN总线设备，控制云台电机和推手电机，采用双环PID算法，控制两个电机的位置。
   4. 两个电磁阀，控制抓手的升、降、开、合。
3. 逻辑控制
4. 结合控制板2和树莓派3B+回传的信息综合判断自身位置
5. 根据四个CCD摄像头进一步校准自身位置
6. 在合适的位置抓取物料，结合树莓派摄像头的信息判断颜色
7. 规划路径，通过自身储存的路径，根据颜色的不同进行选择。
8. 发送指令给树莓派3B+和控制板2，控制它们的程序。

**主控板2：负责辅助计算**

主要功能，包括：

1. 连接两个辅助轮编码器，使用STM32定时器中的编码器功能记录编码器返回的脉冲数，并且将其向步进电机切向方向分解，充当步进电机的编码器。辅助CCD摄像头的姿态校准。
2. 连接陀螺仪，辅助测定车身姿态。
3. 计算步进电机旋转角度，及时使反馈给控制板1。
4. 减轻控制板的部分计算负担。

**树莓派3B+：负责颜色识别和位置识别**

* 1. 开机自启动python脚本，开启多进程，等待控制板1的指令。
  2. 如收到颜色识别指令，开启摄像头1，进行HSV像素统计，将像素点最多的

那种颜色返回给控制板1。

* 1. 如果收到位置校准指令，则开启摄像头2进行霍夫抓圆，寻找最近的圆圈坐标，返回给控制板1

算法与竞赛策略

1. 底盘采用四轮全向底盘，运用雅可比矩阵算法，将速度分解到各个轮子上。
2. 上层抓手控制采用双环PID算法，两个电机通过CAN总线反回自己的位置速度等信息，通过位置环PID和速度环PID，控制抓手位置。
3. 利用CCD摄像头传感器反回的1X128像素点，识别地图黑线，通过闭环PID完成巡线，位置测定。
4. 利用辅助轮的编码器，给步进电机构建闭环PID，精确控制电机转动。
5. 针对任务1，我们按照1234的顺序抓取物块，按照不同的颜色分别放置，对于在中心的黑色，我们采用最后放置的策略，及放完其他物料再放置黑色。
6. 针对任务2：按照ABCDE顺序抓取物块，抓取ABCD时遇到黑色则搬运到特定位置，遇到别的颜色自主规划路径，根据场地上的黑线特征和摄像头回传的圆圈坐标，判断自己的位置，最后将黑色抓取回到home区。