智能小车详细方案设计

荆尹浩宇 滕一铭 张佳悦

2021年7月9日

目录

[智能小车详细方案设计 3](#_Toc2660)

[1. 机械结构设计 3](#_Toc30362)

[1.1 总体结构及车体详细设计 3](#_Toc28538)

[1.1.1 总体结构设计 3](#_Toc8253)

[1.1.2 设计计算 4](#_Toc32762)

[1.1.3 材料与制造方法 5](#_Toc32290)

[1.2 抓取机构详细设计 5](#_Toc4417)

[1.2.1 机械结构尺寸 5](#_Toc15888)

[1.2.2 设计计算 6](#_Toc6225)

[2. 控制系统设计 7](#_Toc28145)

[2.1 硬件单元详细设计 7](#_Toc27798)

[2.1.1 主控板 7](#_Toc11270)

[2.1.2 执行器 7](#_Toc19913)

[2.1.3 传感器 8](#_Toc20919)

[2.1.4 电源需求 9](#_Toc11775)

[2.1.5 通信方式 9](#_Toc13797)

[2.1.6 电气系统模块 10](#_Toc2343)

[2.2控制算法详细设计 11](#_Toc19188)

[2.2.1 循迹部分 11](#_Toc7557)

[2.2.2 避障部分 12](#_Toc15836)

[2.3 软件详细设计 14](#_Toc13543)

[2.3.1 小车总体软件设计 14](#_Toc9986)

[2.3.2 循迹部分软件设计 15](#_Toc27820)

[图11 循迹部分软件逻辑框图 15](#_Toc8878)

[2.3.3 避障部分软件设计 15](#_Toc6868)

[3. 成员分工与工作计划 16](#_Toc5717)

[3.1 成员分工 16](#_Toc8361)

[3.2 工作计划 16](#_Toc20755)

智能小车详细方案设计

在进行文献调研、需求分析与原理方案设计后，为细化原理方案，总结系统总体方案及各个单元具体设计、部件选型、设计计算、仿真分析成果，并为后续制造加工、编写调试提供理论依据与实践指导，笔者将智能小车的机械结构与控制系统的详细设计内容进行了总结梳理，并对小组成员分工与后续工作进程进行了初步规划，具体如下：

1. 机械结构设计

1.1 总体结构及车体详细设计

1.1.1 总体结构设计

在总体结构上，经过深入调研与设计，我们最终采取的设计方案相比于第一次汇报时的设计方案有了很大的改动。首先是放弃了麦克纳姆轮而采用普通车轮，具体原因有三点：首先是麦轮与地面的硬接触摩擦力较小，采取软质的车轮有利于控制打滑，相应地在转向控制上也具有优势；其次是采用差速法转向相比麦轮在算法上复杂度更低；最后是麦轮驱动需要更大的电机，小车自重与耗能也随之增加，在保证功能实现的前提下采用普通车轮更加合适。之后，我们计划采用两个驱动轮加两个牛眼万向轮的结构，在老师的指导下改成了两个驱动轮加一个万向轮的三轮结构，原因在于四轮小车车轮难以共地会造成轻微颠簸的问题。由于小车机构的限制我们难以做出悬挂机构保证四轮小车车轮均保持接触地面，改为三轮可以保证行驶过程中的稳定性，同时三轮结构也能很好地缩小车身尺寸。在车身分层以及硬件位置安排上，我们基本保留了原方案。

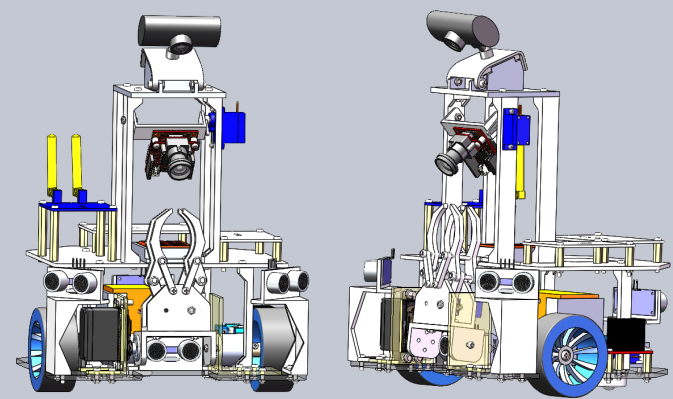


图1 小车3D模型图

最终设计出的车身外轮廓为200mm×200mm的正方形，规整的形状和尺寸较小的车身有利于避障时规避物体，同时也有利于在设计计算时车体的抽象建模。在车架布局上，我们最终决定使用两个普通车轮与一个万向轮，分别置于小车的前端左右两侧和后端。小车总体分为三层，具体布局如上图，第一层主要放置机械臂基座、电源电池组、电机及其驱动模块、超声波传感器等，第二层放置单片机、电压转换模块、IMU、蓝牙模块和WiFi模块等，第三层放置遥控用摄像头、OpenMV及舵机。

在修改为这一版方案后，巡线避障过程中的转向将通过差速法完成，由以下方程可以根据需要的线速度和转弯半径计算得到各轮线速度。

其中、为两轮速度，为车辆线速度（按中轴线计算），为转弯半径、为车辆轮距。在实际程序设计实现下，该算法可以较为灵活准确的控制车辆的转向、行走速度。

1.1.2 设计计算

鉴于IMU惯性测量单元安装在接近重心处性能最好，因此车身重心的确认有一定的重要性。而根据老师的指导可知，寻线测量模块与转向中心重合时更为准确，因此让转向中心与重心相接近有利于其他硬件位置的确定。由于车身大部分硬件重量不大，重心计算主要集中在电池组、舵机、电机与车轮，由于舵机与机械臂结构在车体前方，为了让重心接近转向中心（采取三轮两驱结构之后，转向中心就是两个驱动轮的对称中心），将电池组放置在驱动电机后方，在本次尺寸标定中，将电池安装在几何中心稍稍偏后的位置，而电机、大舵机在几何中心前方，经过近似和力矩平衡计算得出，这样的安排基本符合前后均匀配重的要求。为了保证在抓取、行进过程中车身姿态稳定不翻倒，我们对各种情况下的车身重心的位置进行了近似估算，计算过程如下图。

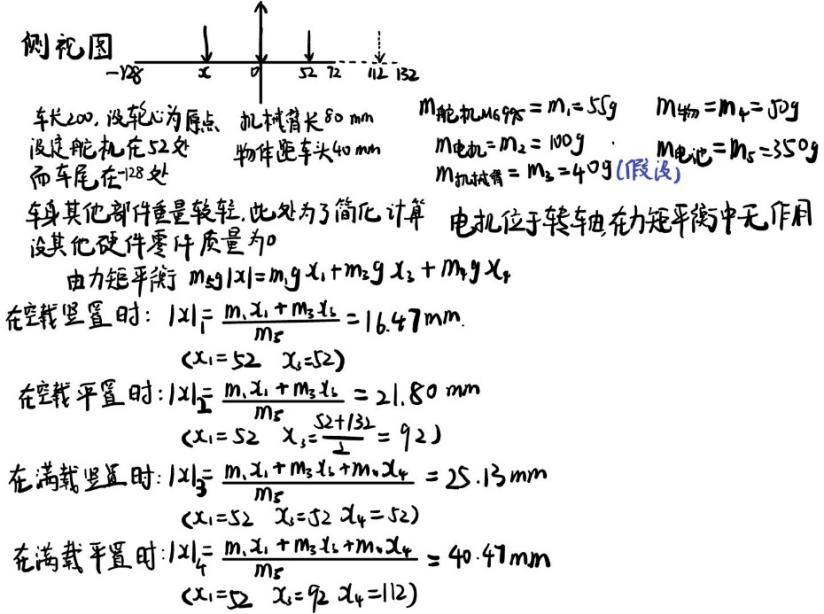


图2 各种情况下车体重心位置的计算

由于三轮小车的第三轮（牛眼万向轮）在小车后方，因此在平衡校核时主要考虑向前倾倒的情况，因此在满载平置的时刻转轴前方压力最大（即上图中x4的情况），综合考虑到其他部件的自重以及对于车身其他运动状况下重心平衡的要求，我们最终确定x=40mm，即电池组的重心距离车位78mm，基本可以满足各种运动情况下小车的平衡。

1.1.3 材料与制造方法

在组成小车的各个零部件中，单片机、OpenMV、超声模块等传感器、直流减速电机、舵机等执行器、驱动模块、ST-Link、联轴器、电源、螺钉紧固件等均可以直接购买；各层底板均为亚克力板，通过激光打印加工；其他固定用零部件为尼龙件，采用3D打印制造。

1.2 抓取机构详细设计

1.2.1 机械结构尺寸

抓取结构的设计上，我们基于原有方案在结构上进行了优化，使其轻量化且体积较小。我们最终选用的机械臂有两个自由度，用两台舵机控制手爪开合与机械臂旋转角度完成抓取。具体而言，机械臂的转动轴与小车横向平行，抓取货物时在小车前方，将货物装上小车后旋转至小车结构中，避免了对于摄像头收集图像信息的干扰。在具体的结构设计上，对于手爪的每个爪体用两个转轴连接，实现转动完成抓取。上部的轴一端固定，一端与手爪连接，下部轴一端与手爪连接，一边为齿轮，两侧由一个舵机驱动，通过齿轮传动同时驱动两个手爪。机械臂整体较为小巧，抓取货物后长度约为10cm，结构如下图所示。

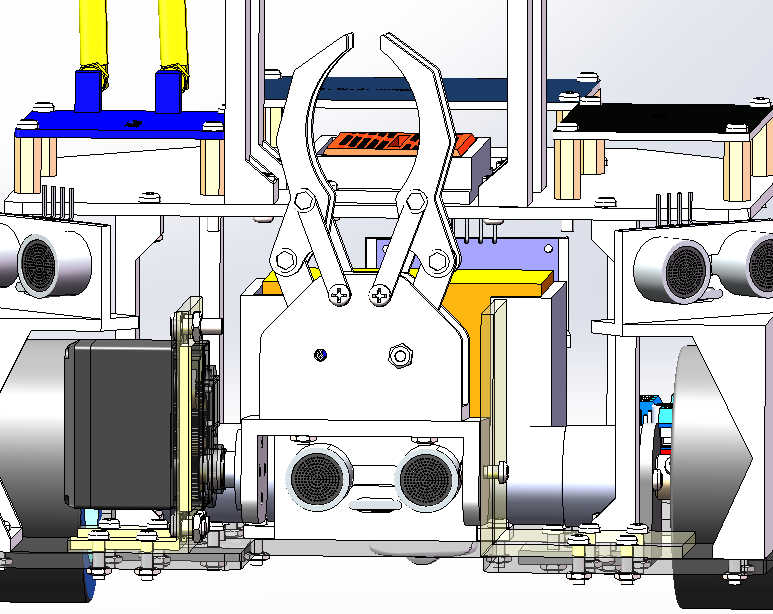


图3 机械手爪结构设计

1.2.2 设计计算

抓取结构的平面设计图如下图左侧，可以将右侧受舵机驱动的抓手简化为下图右侧所示。抓取机构总体尺寸比较小巧，因为抓取的物体很轻，因此对于材料承压能力要求不高，因此除了必要的螺钉紧固件，其他非标部件均采用3D打印尼龙件，成本低且制造方便。抓取部分用一个SG90舵机驱动齿轮转动，带动手爪完成开合动作。抓取机构安装在一块与MG995连接的C型板上，通过MG995控制整个机械臂在竖直平面内的转动。

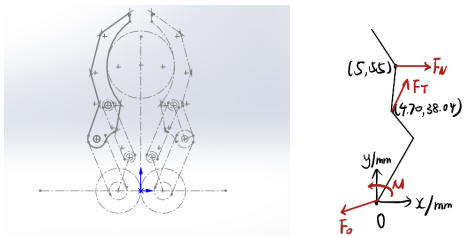


图4 抓取结构的平面设计图及计算简化模型

抓取机构的力学校核具体如下：考虑初始抓取时刻的情况，即对抓取机构摩擦力要求最高的情况，此时摩擦力方向垂直于xoy平面，可不考虑；而此时力FN平行于x轴，FT在xoy平面内的分量为0。此时若将机械手爪模型简化为刚体，则可通过力矩平衡求出手爪对物体的压力。

已知SG90舵机在5V下的转矩为1.5kg·cm，由于需要同时驱动两个手爪，根据对称性可知上述力学模型中的M=0.75kg·cm=7.35N·cm，由力矩平衡得FN=M/y=7.35/5.5=1.34N。根据已知条件物品质量50g，受到重力0.49N，若要满足可以夹起的条件，单个手爪必须能够提供大于0.245N的最大静摩擦力，由约束条件可知，μ≥0.183。由此可得，只要手爪与物体接触面不过于光滑，就可以实现在初始时刻顺利夹取物体的操作。在后续机械臂抬升的过程中，垂直xoy平面的重力分量不断减小，对摩擦力要求逐渐降低，因此SG90 满足使用要求。

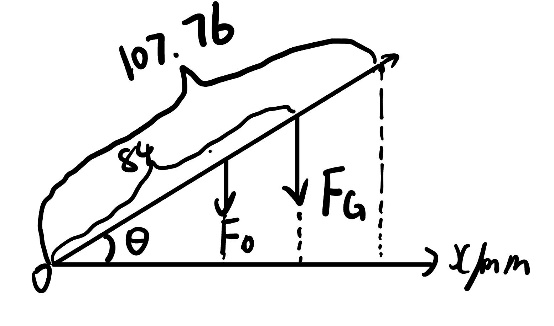


图5 机械臂受力情况简图

MG995舵机的校核如下，在抬升过程中，机械臂受力可简化如上图。为了简化计算，同上假设机械臂的总重量40g（大于实际重量），并假设机械臂的重心在整个臂长的中点，易知在θ=0时F0与FG对o点的合力矩最大，在不考虑摩擦力的状况下Mmax=FG×lG+F0×l0=0.96kg·cm，远小于MG995舵机在5V下的工作扭矩13kg·cm，在零件3D打印的条件下摩擦力的大小也在合理范围内，因此MG995舵机已经足够使用。

1. 控制系统设计

2.1 硬件单元详细设计

2.1.1 主控板

在我们的方案设计中，对主控板的需求主要包括基本的数据运算、生成PWM信号、基本I/O、串行通信与数据收发等等，且对于算力有一定的要求。由于智能小车中主控板需要控制比较多的硬件，因此对于引脚数量需求较高。此外，还应在满足上述条件的基础上尽可能减小尺寸、降低成本。

综合上述分析，我们决定选用stm32F103VET6开发板作为主控板，它是一个32位高密性能微控制器单元，有三个12位模数转换器和4个通用16位计时器，外加两个PWM计时器，同时具有标准和先进的通讯接口，高达两个IC，三个SPI和两个IS，一个SDIO，五个USART，一个USB和一个CAN。它融合了高性能的ARM®Cortex®-M3 32-位RISC内核，运行频率72MHz，高速内嵌存储器和一个宽范围的强化输入输出外设，连接至两个APB总线。此外，它还具有512kB闪存。总体而言，该单片机功能强大，引脚、接口资源丰富，可以满足需求。

关于stm32与其他硬件的连接，包括直接引脚连接，借助蓝牙模块与移动设备无线连接，通过ST-link与电脑USB连接（用于单片机的调试和程序的烧写）等，目前已有的设备能够满足上述连接方式，因此选用该单片机能够满足需求。

2.1.2 执行器

本次任务所需的执行器主要包括电机、舵机及驱动器等，下面将分别从舵机驱动能力、电机驱动能力和摩擦力校核上进行分析设计并验证可行性。

首先是舵机驱动能力分析。SG90舵机的工作电流大约为300mA，而MG995舵机的工作电流约为100mA。单个舵机的功率一般不高，可以通过单片机的引脚输出PWM对于舵机直接进行控制，电流足够。

其次是电机驱动能力分析。我们选用的是25GA370带编码器减速直流电机，空载转速为280rpm，电机额定电压为12V，电流为0.3A，可得出额定功率最大为3.6W。电机需要的电流远大于单片机IO接口输出的电流（通常为8~20mA），因此需要使用专门的电机驱动模块。而课程提供的L298N的最大功率为25W，因此一个驱动器驱动两个电机的方案完全可行。电机额定转矩为0.4kg·cm，即3.92N·cm。我们此次选用的是直径为65mm的车轮，由此进行计算，在地面不打滑的情况下，电机可以为小车提供2.41N的驱动力，可以满足本次任务的驱动需求。

最后是摩擦力校核。为简化讨论，可以假设小车质量为1kg（实际应该大于该值，这里为了饱和式校核选择较小的质量）。若要不打滑，即要求两个轮子都要满足纯滚动条件，即切向加速度等于角加速度乘以半径，角加速度为（此式中将相对轮心的转动惯量计算模型做了简化）。将计算结果乘以半径0.065m后得到加速度a=4.825m/s2。设地面与轮胎间动摩擦因数为μ，由平衡方程故，因此对于场地的粗糙程度有一定的要求，但μ=0.49依然属于相对较为光滑的表面，经过我们对场地的考察，当前的车轮表面与场地条件基本可以满足这一条件。

2.1.3 传感器

本次任务所需要的传感器主要包括超声传感器、OpenMV视觉模块、JY901S惯性测量单元，下面将分别具体分析三者的功能及在任务中发挥的作用。

首先是超声传感器。老师为我们提供的是HY-SRF05超声传感器，与我们原本选择的HC-SR04在引脚与尺寸上存在一定区别，但是二者的基本功能及工作原理相同，因此可以在避障中发挥所需的作用。两个器件的基本性能参数完全一致，工作电流均为15mA，直接接上电源模块的5V电源和单片机的相应引脚即可正常工作。传感器的感应角度不大于15度，因此需要在小车车身周边安装多个超声传感器来满足避障的探测需求，尽量增大检测范围。两者的探测距离均为2cm-450cm，根据本次场地的大小以及避障的基本需求，两款超声都可以满足需要。此外，两款传感器都有可达0.2cm的高精度，有利于小车的精确避障。

其次是OpenMV视觉模块。OpenMV采用的STM32H743核心主频可达400MHz，理论上每秒处理的帧数可达300帧，其CPU的RAM达到了256KB，而一帧灰度图像的大小为72KB，运算能力强大，可以满足循迹避障需要的图像识别与计算的需求。它的全速 USB (12Mbs) 接口连接到电脑时会出现一个虚拟COM端口和一个“U盘”，与电脑连接可进行实时图像传输和程序烧写。它的SD卡槽拥有100Mbs读写，这允许OpenMV摄像头录制视频并把机器视觉的素材从SD卡提取出来，但由于本次任务中只需要自主巡线和避障，对于视频存储没有很高的要求，因此只需要考虑OpenMV在图像处理上的性能即可。

最后是JY901S惯性测量单元（IMU）。模块内部自带电压稳定电路，工作电压3.3v~5v，引脚电平兼容3.3V/5V的嵌入式系统，连接方便。模块内部集成了姿态解算器，配合动态卡尔曼滤波算法，能够在动态环境下准确输出模块的当前姿态，姿态测量精度静态0.05度，动态0.1度，稳定性极高。支持串口和IIC两种数字接口。方便用户选择最佳的连接方式。串口速率2400bps~921600bps可调，IIC接口支持全速400K速率。

2.1.4 电源需求

任务所用的各硬件模块的电源需求整理如下表，LM2596的引脚数目与负载电流大小3A能够满足目前的用电需求。

表1 各硬件模块电源需求

|  |  |  |
| --- | --- | --- |
| 硬件 | 数量 | 所需电源电压（V） |
| Stm32F103VET6单片机开发板 | 1 | 5 |
| 电机编码器 | 2 | 3.3 |
| L298N驱动模块 | 1 | 12 |
| MG995舵机 | 1 | 5 |
| SG90舵机 | 3 | 5 |
| HC-SR04/HY-SRF05超声传感模块 | 4 | 5 |
| HC-05蓝牙模块 | 1 | 5 |
| JY901S九轴IMU惯性测量模块 | 1 | 3.3 |
| OpenMV机器视觉模块 | 1 | 5 |
| WIFI图传 | 1 | 5 |

2.1.5 通信方式

各个硬件之间的通信方式大致分为如下三种：IMU、openMV与单片机通过串口通信，舵机、超声波传感器、电机编码器与单片机通过IO功能与内部协议传输信息，网络摄像头与PC通过wifi进行图传。

2.1.6 电气系统模块

STM32的引脚资源可以满足任务的需求，系统的电气原理图及电气连线表如下：

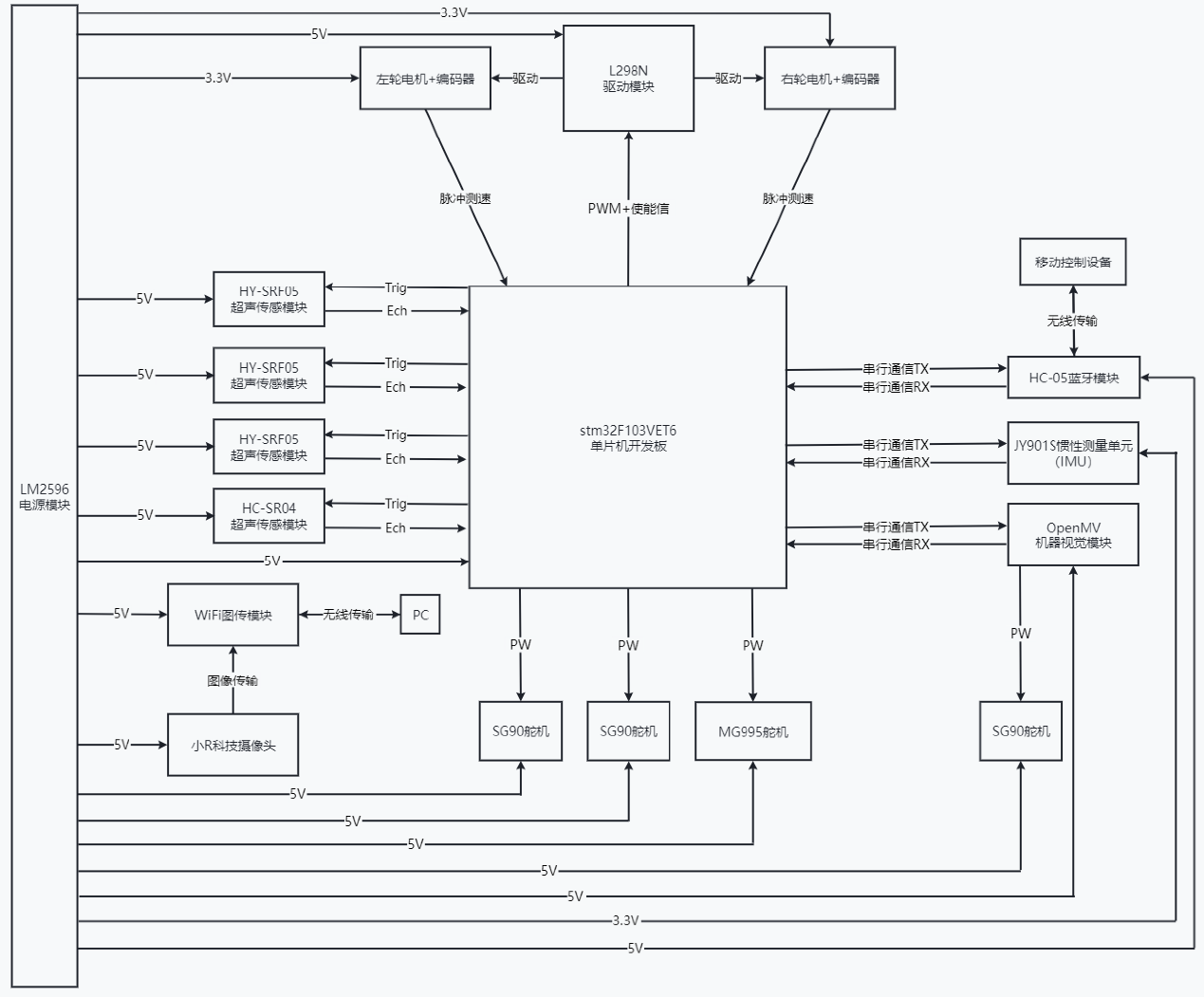


图6 系统电气原理图

表2 系统电气连接表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块 | 硬件 | 端口 | 引脚编号 | 引脚定义 |
| 驱动控制 | L298N | IN1 | PE1 | I/O |
| IN2 | PE2 | I/O |
| IN3 | PE3 | I/O |
| IN4 | PE4 | I/O |
| PWM | L298N | ENA | PA2 | TIM5\_CH3 |
| ENB | PA3 | TIM5\_CH4 |
| MG995 | 信号线 | PE9 | TIM1\_CH1 |
| 机械臂SG90 | 信号线 | PE11 | TIM1\_CH2 |
| 超声SG90 | 信号线 | PE13 | TIM1\_CH3 |
| openMV | openMV SG90 | 信号线 | P7 | TIM4\_CH1 |
| 脉冲测速 | 左轮电机编码器 | A | PA0 | TIM2\_CH1 |
| B | PA1 | TIM2\_CH2 |
| 右轮电机编码器 | A | PA6 | TIM4\_CH1 |
| B | PA7 | TIM4\_CH2 |
| 超声测距 | HC-SR04(前) | Trig | PE5 | I/O |
| Echo | PC6 | TIM8\_CH1 |
| HC-SR04(左前) | Trig | PE6 | I/O |
| Echo | PC7 | TIM8\_CH2 |
| HC-SR04(右前) | Trig | PA4 | I/O |
| Echo | PC8 | TIM8\_CH3 |
| HC-SR04(后) | Trig | PA5 | I/O |
| Echo | PC9 | TIM8\_CH4 |
| 串行通讯 | OPENMV | TX | PA9 | USART1\_TX |
| RX | PA10 | USART1\_RX |
| HC-05 | TX | PC12 | UART5\_TX |
| RX | PD2 | UART5\_RX |
| IMU | TX | PB10 | USART3\_TX |
| RX | PB11 | USART3\_RX |

2.2控制算法详细设计

2.2.1 循迹部分

循迹部分所用的传感器主要为OpenMV和IMU，需要进行图像处理从而确定道路类型。图像处理的过程如下：首先需要通过OpenMV获得道路灰度图像，其次需要进行二值化与滤波处理，之后需要提取有效信息，得到小车所在位置与道路中点的偏移量。

从收集图像到得到偏移量的处理过程会具体应用在下面的算法中（如下图所示）。小车在抓取货物后，从OpenMV检测到与两条道路黑线相交的起点线开始进入循迹算法。在小车前进过程中，需要通过OpenMV获取图像，并在二值化及滤波处理后找到底部有效行，计算出道路中点与小车所在位置的偏移量，并将该信号反馈给PID控制函数pos\_pid=KPR×（a-b）+KIR×a+KDR×(a-2b+c)，a、b、c分别为过去三次的偏差量，KPR、KIR、KDR分别为比例常数、积分常数、微分常数，以调节小车速度及前进方向。

程序结束的标志在于小车是否检测到终点线，即一条笔直的与两条道路黑线相交的黑线，若检测到终点线，则小车减速停止，循迹算法结束；若未检测到终点线，则小车继续运动，重复上述获取信息并调节运动速度及方向的过程。

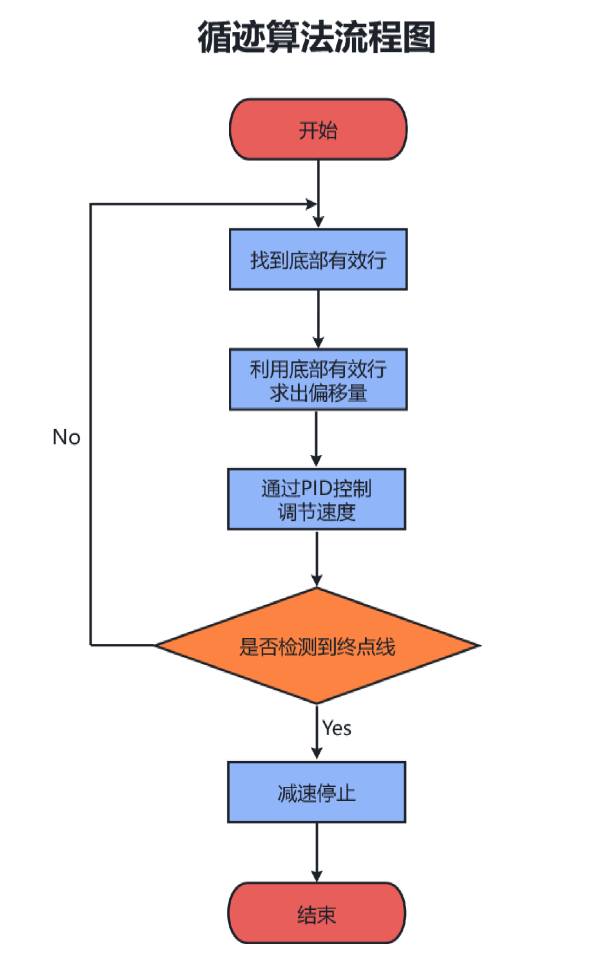


图7 循迹算法流程图

OpenMV中有许多系统内置的图像处理函数，从一定程度上保证了算法的可实现性。但算法的实现过程也可能出现诸多问题，比如可能存在出现盲区的问题。在弯道行驶的过程中，若小车速度较快，弯道过急，会导致小车上的OpenMV在进入弯道时不能完整地检测道路的有效信息，即出现小车进入盲区的问题。这可能使小车在一段时间内无法为控制器提供被控量，进而使整个控制系统崩溃，小车偏离轨道。因此，可以考虑将小车的速度减小程度与测得的道路斜率相关联，避免出现这种问题。其他类似的可能出现的问题也需要在后续的调试过程中不断细化和优化算法。

2.2.2 避障部分

避障部分的算法如下图所示，首先通过IMU确定小车的初始位姿与初始行进方向，在小车开始运动后，利用超声传感器不断测量障碍物距离小车的距离，并判断该距离是否小于规定的最小距离。若不小于规定距离，则小车保持直行，继续进行检测；若小于规定距离，则需要通过人工势场法与随机法的叠加计算出小车新的行进方向，使小车转向。在OpenMV检测到终点线（一条笔直的黑线）之前，不断重复进行上述的测量并迭代计算出新方向的过程，如果OpenMV检测到终点线，则小车在目标的引力场与围栏的斥力场下到达终点附近，并通过OpenMV寻找放置点，避障算法至此结束。

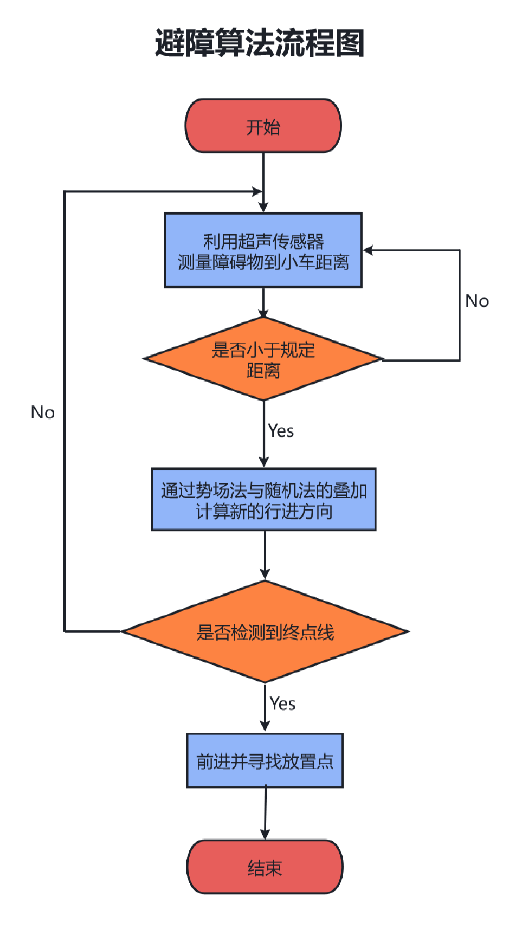


图8 避障算法流程图

在算法处理的过程中，迭代求解小车运动新方向的过程具体如下。首先是通过IMU确定小车位置，并通过超声确定小车前方左中右三侧距离障碍物的距离，二者是小车需要接受的主要数据。其次是通过建立目标点的引力场及障碍物与围栏的斥力场，确定小车周围方向的势能高低，从而选择最低势能方向。具体来说，可以在小车的周围画出一个小半径的圆形，在其上取出八个点，通过计算比较八个点中势能最小点，并以小车所在位置到该点的方向为下一步方向。同时，还需要通过随机算法确定方向，即小车前方的三个超声传感器中，左侧传感器检测到某个距离内的障碍物则右转，右侧传感器检测到障碍物则左转，正前方传感器检测到障碍物则向任意方向选择90度，左方及前方、右方及前方检测到障碍物则同理改变方向。最后需要将人工势场法得到的方向与随机法得到的方向进行加权，由此获得最终小车前进的方向。

利用人工势场法与随机算法加权获得方向会有两点好处：一是可以在一定程度上避免人工势场法遇到的局部最优解及障碍物与目标点过近时无法到达目标点的问题，二是考虑到人工势场法利用IMU获得绝对位置计算势能会存在明显的误差，随机算法可以在一定程度上加快算法迭代过程并使运动过程更合理。

下图为对于避障部分的算法仿真，在对场地与运动过程进行一定的简化和抽象后，我们利用上述算法得到了小车的运动过程。图中红色星形为起始点与目标点，蓝色圆形为障碍物，绿色轨迹即为小车的运动轨迹。仿真结果在一定程度上验证了目前算法的可行性，但具体的加权值的大小与IMU的精度相关联，且IMU控制与超声测距存在的问题也需要在调试过程中不断改善。

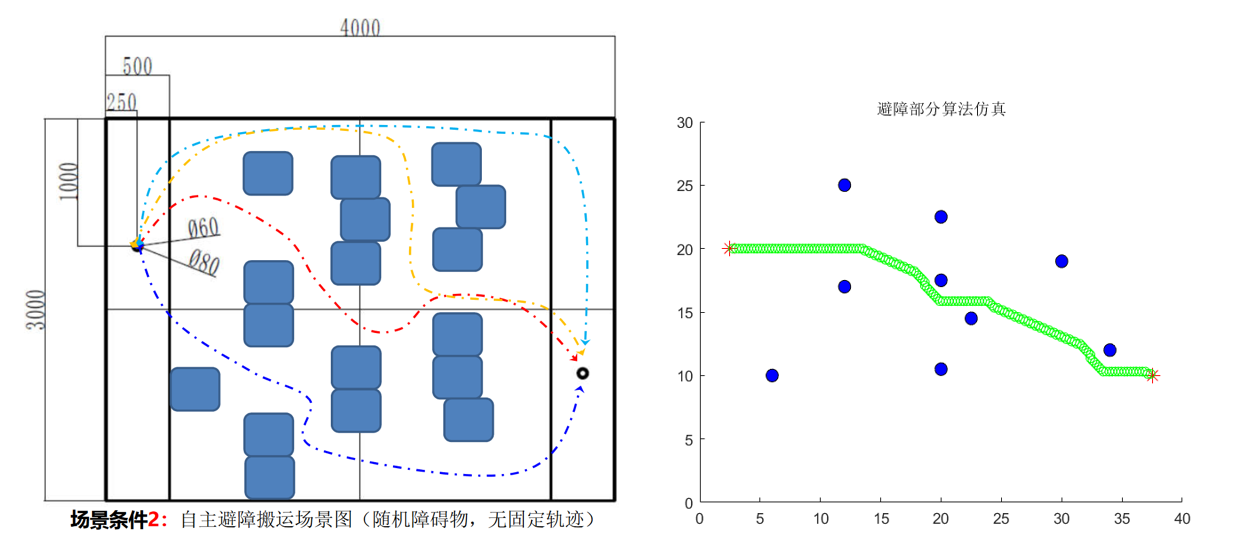


图9 避障部分算法仿真

2.3 软件详细设计

2.3.1 小车总体软件设计

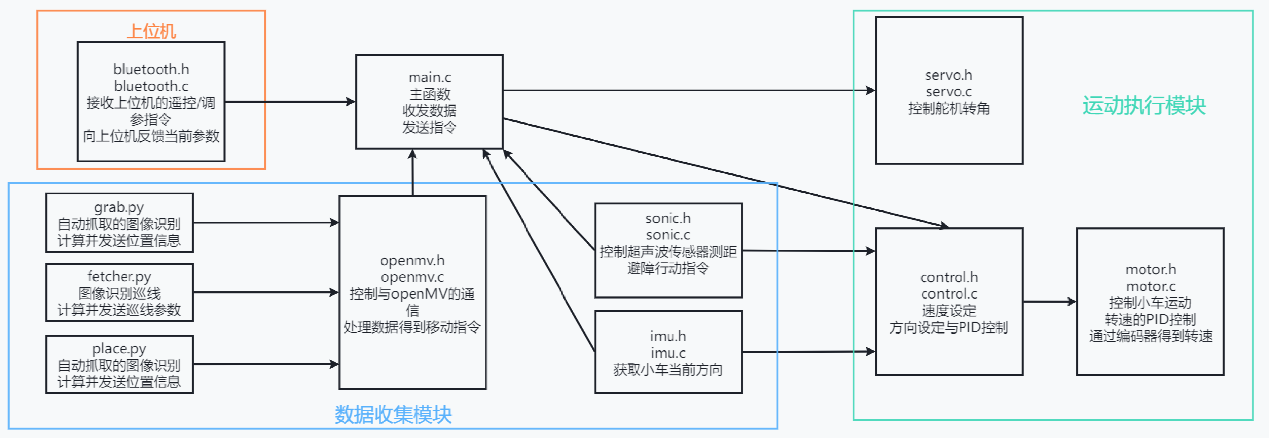


图10 系统软件逻辑框图

系统软件模块可分为上位机、数据收集、运动执行三个部分，上位机模块主要包含蓝牙函数，负责手机与单片机的通讯，完成参数调试、遥控的任务，数据收集模块包含OpenMV的相关函数，以及超声波、IMU惯性测量模块的函数，负责包括路线、障碍物、车身方位等信息的搜集。这些数据传入单片机后在main函数中进行处理，之后运动执行模块通过PID控制改变小车车轮的速度与方向。在小车自动抓取放置物品时，与openMV直接相连的舵机在OpenMV的控制下转动调整视角寻找目标点位，再进一步控制下位机进行抓取与放置。

软件模块的全局变量包括小车前进的速度与方向、舵机转角等等，具体的函数以及中介变量会在后续编写代码的过程中进一步细化。

2.3.2 循迹部分软件设计

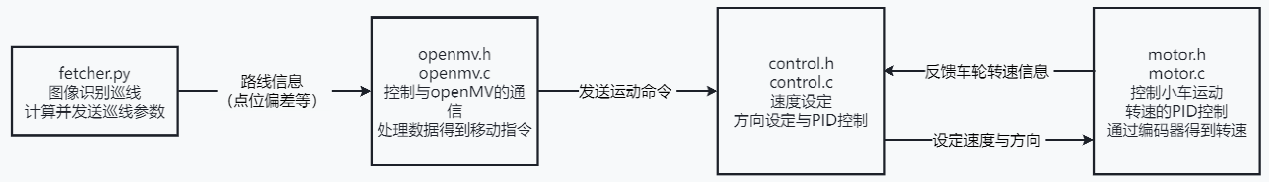


图11 循迹部分软件逻辑框图

循迹部分的逻辑相对较为简单，考虑到车轮转速的PID控制和编码器反馈的速度紧密关联，我们将速度PID的函数集成在motor子程序里，而control子程序主要进行小车总体行进过程中的速度设置和方向设置。具体而言，小车前进过程中，接收到openMV传入的路线信息后，单片机在openmv子程序中对数据进行必要的处理之后，通过control模块将小车的目标运行速度与方向转化为两个车轮的转速，通过motor模块控制两轮按照相应转速运动，从而实现循迹功能。

2.3.3 避障部分软件设计

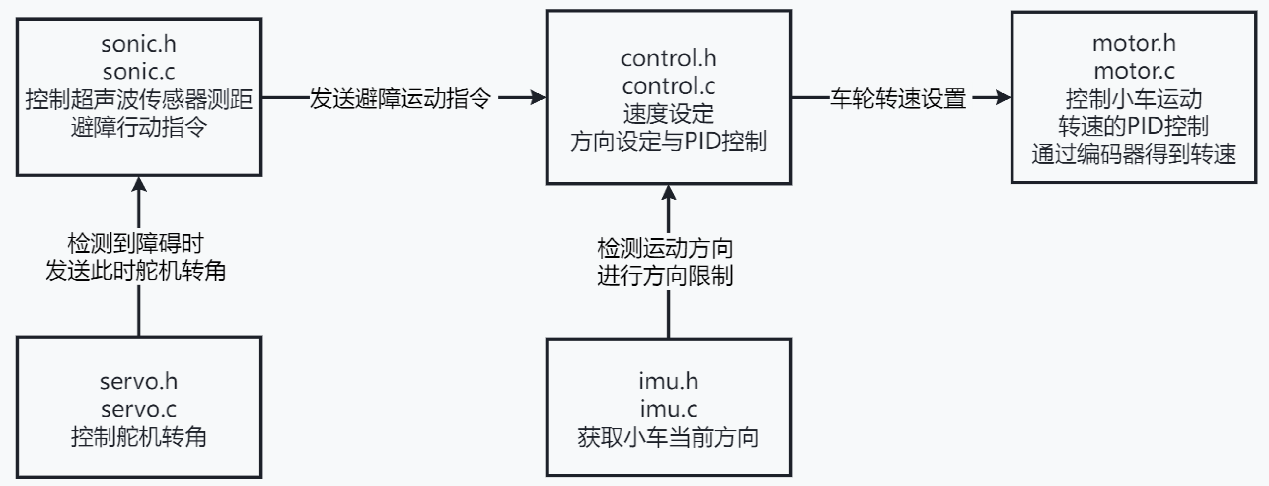


图12 避障部分软件逻辑框图

避障部分的逻辑则相对复杂一些，因为其既涉及到小车和周边障碍物的位置关系，还需考虑小车车体的方向。小车的前方有三个固定的超声传感器，在sonic子程序控制下以固定频率向单片机传输各自方向障碍物的距离，尾部的一个传感器在小车向后运动时则在舵机的控制下以固定频率左右旋转，将不同方向的障碍物距离数据传输给单片机。单片机得到数据之后通过sonic模块中的函数运用避障算法给出行动指令，单片机通过control模块将小车的目标运行速度与方向转化为两个车轮的转速，通过motor模块则控制两轮按照此转速运动，从而完成巡线。此外，还有imu子程序不断通过串口读取车身方向的数据，通过imu中的函数对小车的前进方向做出一定的限制，进而通过control模块改变小车速度及方向，确保小车可以到达终点。

1. 成员分工与工作计划

3.1 成员分工

经过小组同学协商，组内成员的大致分工如下：

表3 详细设计及后续工作成员分工

|  |  |  |
| --- | --- | --- |
| 分工 | 主负责人 | 具体情况 |
| 车体结构 | 滕一铭 | 荆尹浩宇辅助进行尺寸设计，三人共同建模装配 |
| Stm32下位机控制与电气系统 | 荆尹浩宇 | 张佳悦辅助进行软件设计，三人共同编程调试 |
| OpenMV相关控制与硬件通讯 | 张佳悦 | 滕一铭辅助进行算法设计与仿真，三人共同编程调试 |

3.2 工作计划

为明晰阶段任务，提高合作效率，我们将智能小车的总任务进行分解，并制定以下计划：

表4 后续工作计划表

|  |  |
| --- | --- |
| 时间节点 | 阶段任务 |
| 7月8日-7月10日 | 进一步完善详细设计方案，采购零件，编写部分硬件单元代码 |
| 7月11日-7月13日 | 加工零部件，组装小车硬件，编写部分硬件单元代码 |
| 7月14日-7月15日 | 完成单元控制代码编写与硬件测试 |
| 7月16日-7月17日 | 硬件通讯部分代码编写与调试，软件集成 |
| 7月18日-7月19日 | 自动抓取货物与放置货物集成调试与细节优化 |
| 7月20日-7月21日 | 循迹模式小车运动集成调试与细节优化 |
| 7月22日-7月23日 | 避障模式小车运动集成调试与细节优化 |
| 7月24日-7月25日 | 小车整体最终调试，准备比赛 |

以上计划依课程时间计划表及小组成员初步讨论制定，若各部分任务实现过程中遇到问题，允许依具体情况适度改变。