

上海交通大学

工科创IVE 课程报告

组 名:	PSoC 第六组
组 长:	张沛东 516021910178
组 员:	陈思哲 516021910038
组 员:	牟天昊 516030910191
组 员:	忻旻健 516030910168
组 员:	叶健龙 516030910172

2018-2019 学年第 2 学期

2019 年 5 月 5 日

目录

一 实验概述	1
1.1 实验平台与模板	1
1.2 实验目的	1
1.3 实验任务	1
1.2.1 巡线任务	1
1.2.2 迷宫任务	1
1.2.3 自选任务	1
二 实验任务详解	2
2.1 巡线任务	2
2.1.1 任务描述	2
2.1.2 设计思路	2
2.1.3 算法实现	2
2.1.4 遇到的问题及解决过程	3
2.1.5 结果与反思	3
2.2 迷宫任务	4
2.2.1 任务描述	4
2.2.2 设计思路	4
2.2.3 算法实现	5
2.2.4 遇到的问题及解决过程	6
2.2.5 结果与反思	6
2.3 自选任务	7
2.3.1 任务描述	7
2.3.2 设计思路	7
2.3.4 算法实现	7
2.3.5 遇到的问题及解决过程	9
2.3.6 结果与反思	9
三 小组成员分工	10
四 小组成员个人总结	10
4.1 张沛东个人总结	10
4.2 陈思哲个人总结	10
4.3 牟天昊个人总结	11
4.4 忻旻健个人总结	11
4.5 叶健龙个人总结	12
五 致谢	12

一 实验概述

目前,随着自动控制等技术以及相关产业的发展,机器人在人类的生产生活中发挥着越来越重要的作用。传统的机器人可以运行预置的程序来完成固定的流程,也可以结合传感技术或人工智能技术进行自主的动作,以满足特定需要。本次实验课程中,我们将利用所学知识,在实践中对机器人的设计和控制有进一步了解。

1.1 实验平台与模板

本次课程实验基于 PSoC4 机器人控制实验平台展开,该控制平台开放 AD, IO, PWM, UART, I2C 等通用接口,以及碰撞、红外测距、超声测距、编码器、加速度计、陀螺仪等传感器接口,可实现传感器、运动控制以及机器人控制任务实验。

实验又以双轮差速机器人为模板,该机器人集成伺服电机、电池、电源模块,以及碰撞、红外测距、超声测距、巡线、编码器、加速度计、陀螺仪等传感器,实验者可用该机器人直接验证控制、通讯等算法,开展实践。

1.2 实验目的

- (1) 熟悉机器人的构造及功能,能对车体及传感器进行有效的安装、拆卸与维修;
- (2) 熟悉 PSoC Creator 软件的应用,能够参照教程,通过软件进行简单的编程以及实现端口和模块的对应分配;
- (3) 熟悉 C 语言编程及 PSoC 内置函数,能够参照教程,通过编程实现测量值读取、电机驱动等功能;
- (4) 根据所学知识有效组合各模块功能,以实现创新;
- (5) 在机器人设计及控制过程中提高自行分析和解决问题的能力。

1.3 实验任务

本次实验分三个小任务,包括巡线任务,迷宫任务以及自选任务。

1.2.1 巡线任务

机器人应能够沿着地面上与地面背景色不同颜色、具有一定宽度的线形轨迹前进。同时在行进过程中应保持一定的速度和流畅性。本次实验中线形轨迹为黑色。

1.2.2 迷宫任务

机器人应能够走出由直立板墙搭建的迷宫。走出迷宫的速度应尽量快,同时过程中避免车体与板墙的碰撞。

1.2.3 自选任务

小组基于此平台与机器人模板,自行设计任务,并自选功能模块完成。经过讨论小组实现如下功能: 1. 实现通过安卓APP蓝牙模块接收信号对机器人进行运动控制 2. 通过电脑蓝牙信号对机器人行走环境进行地图绘制。

二 实验任务详解

2.1 巡线任务

2.1.1 任务描述

机器人应能够沿着地面上与地面背景色不同颜色、具有一定宽度的线形轨迹前进。同时在行进过程中应保持一定的速度和流畅性。

本次实验中线形轨迹为黑色，与地面背景色相差显著，轨迹宽度约为两厘米。轨迹形状为一个不规则环形，存在连弯、急弯的情况。

2.1.2 设计思路

本任务主要应用三个巡线传感器的测量以及 PWM 电机驱动。根据左、中、右三个巡线传感器的测量值，可以大致判断机器人前进方向与轨迹的关系，根据不同情况进行调整。

流程图如图 2-1 所示。

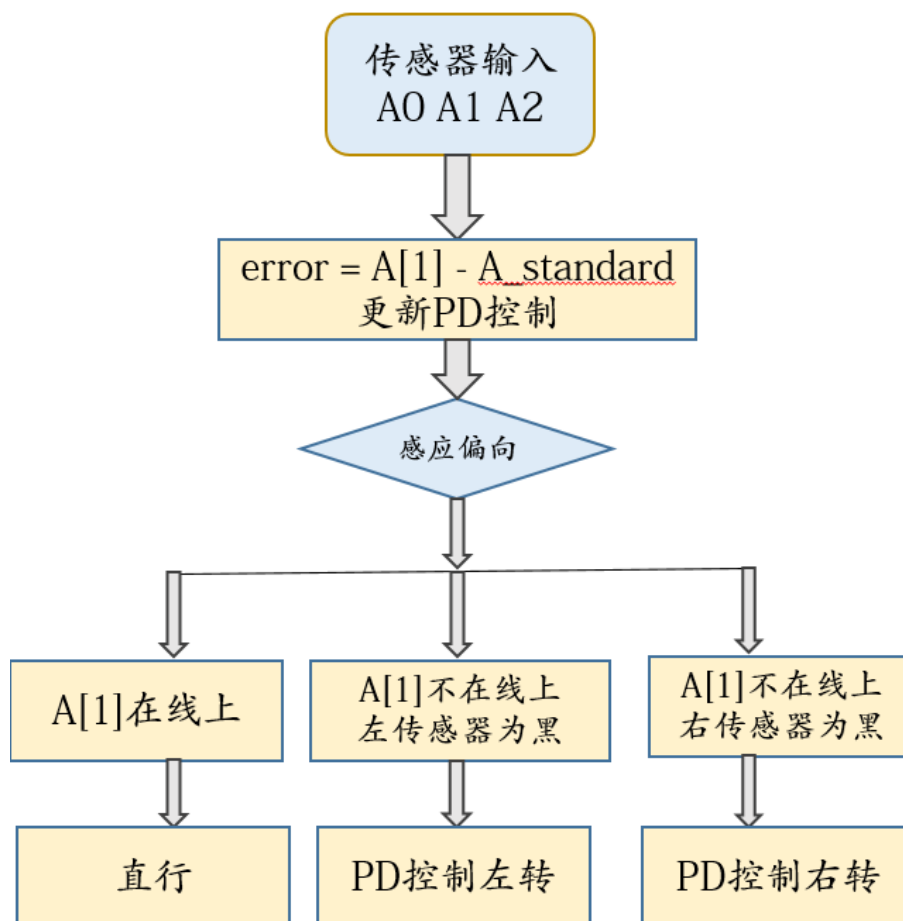


图 2-1 巡线任务流程图

2.1.3 算法实现

(1) 巡线传感器的信号处理

经过实测我们发现，巡线传感器检测到黑色轨迹后输出的模拟量在 500 左右，检测到地面背景色时输出在 1500 左右。我们做出如下判定：A1（中间）传感器的数值低于我们提前设置的阈值 threshold 时则认为偏离了方向。

（2）方向判定

通过反馈三个巡线传感器的亮度值，正常情况下中间传感器的数值低于我们提前设置的阈值 `threshold`，而当该数值突破阈值时，检测左右两个传感器的数值，我们认为较小的一方正在采集到黑线，也即小车车身正往该方向倾斜。在此情况下我们控制小车往反方向转向。

具体表现为：如果中间的传感器 `A1` 在线上，则直行；如果 `A1` 大于某个值 `threshold`，则根据 `A0, A2` 判断车偏左还是偏右，在右边/左边减去一定的修正速度，修正速度由 `A1 threshold` 进行 PD 控制。

（3）电机转速 `PWM_WriteCompare()` 的设置实现

电机借由 `PWM_WriteCompare()` 函数实现，参数为 PWM 值。修复了电机转速不均匀的问题后，测出了两电机的阈值，即不动时的控制值，为 4550。

2.1.4 遇到的问题及解决过程

（1）在测试硬件时发现给定 PWM 后小车的一轮转动均匀，但是另一个轮子却有卡顿的现象，即转一下顿一下。在直行中便表现为存在很大的晃动。

解决：研究发现是右轮轮平面与转动轴不垂直，导致轮子转动时会和框架钢板上的铆钉发生周期性摩擦。我们借用了螺丝刀组卸下了轮胎分析内部结构，将小车右侧的框架钢板卸了下来，对轮子进行调整之后重新安装。

（2）转向实现时以右转向为例，开始的思路是让左轮加速实现向右拐，但是发现这样运行起来车的稳定性不好，将常会巡线失败。

解决：我们选择减慢右轮速度以实现右转向。在这样的转向思路控制下，我们的直行采用当前环境下的较为激进的一组参数。

2.1.5 结果与反思

场地测试的结果令人满意，多次反复实验均能良好地完成巡线任务。如视频演示中那样，巡线连贯且较为精准，达到了我们的预期。

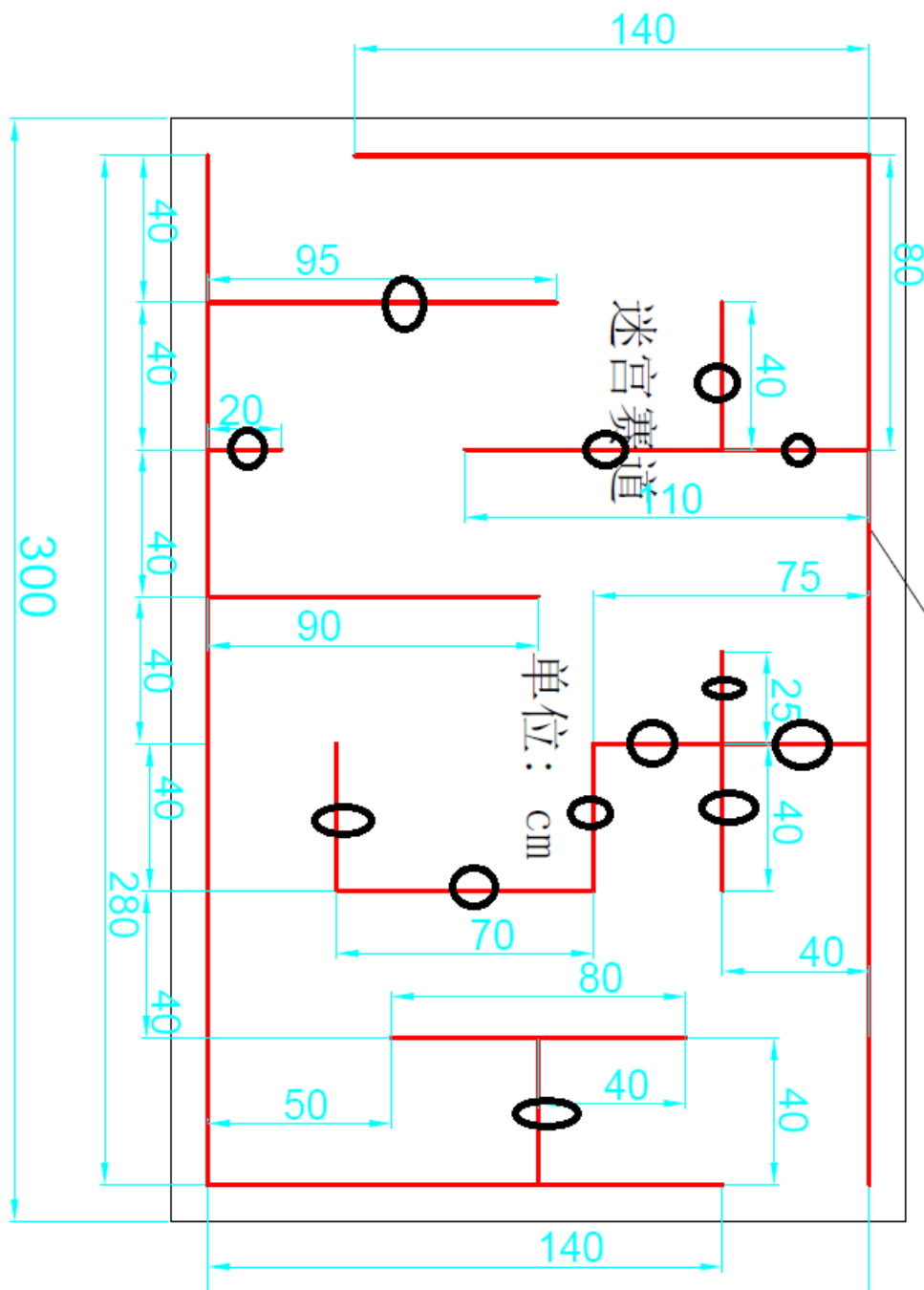
因为在自行设计的场地上尝试了更加急转的弯道，实验场地的巡线便显得游刃有余。这件经历告诉我们，不要满足于最低标准：如果测试时对自己有更高的要求，最后就一定能够得到最好的结果。

值得注意的是我们在实现过程中也尝试了“更多的 if”，即更加复杂的逻辑调整。比如区分大弯和小弯，预设完全跑出黑线等情况，我们预想会有极大的作用然而实际测试中却并不尽人意，添加后没有什么大的改观甚至表现更差了。结果带给我们的启发是最好的不一定是复杂的，调整过参数后的 PD 算法表现出了最好的效果，参数的调试也重要无比，研究过程中的相互鼓励与耐心必不可少。

2.2 迷宫任务

2.2.1 任务描述

机器人应能够走出由直立板墙搭建的迷宫。走出迷宫的速度应尽量快，同时过程中避免车体与板墙的碰撞。



2.2.2 设计思路

对于一个二维迷宫，只要从入口处开始贴着左或右壁行走，最终一定可以走出迷宫。我们选择了贴右壁，并运用了机器人右前方的红外传感器，以及前方的两个碰撞传感器，并将超声传感器固定在小车右方，下面进行详解：

我们考虑小车采取一个简单的从状态映射到动作的策略。状态即为右前方红外传感器、

右方超声传感器，碰撞传感器在某一时刻的数值。动作为左右电机的转速，抽象到高一层次看，可概括为右转，左转，沿墙几类。

对应不同的状态，设置不同的动作，即为我们的设计思路。具体分解如下：

- (1) 情况 1：碰撞传感器返回数值。对应动作：倒车，然后原地差速左转。
- (2) 情况 2：超声传感器返回数值超过一个阈值（对应转角）。对应动作：右转，直到右前方红外传感器返回数值小于一个阈值。实际中我们在此处应用了采样投票算法，后面详解。
- (3) 情况 3：未发生情况 1 或情况 2，正常靠墙行走。对应动作：以一定的超声传感器基准值，用 PID 控制小车靠墙前进。如果小车的偏离过大，则再附加一个快速修正的量。

我们对该流程的设计如图 2-2。

2.2.3 算法实现

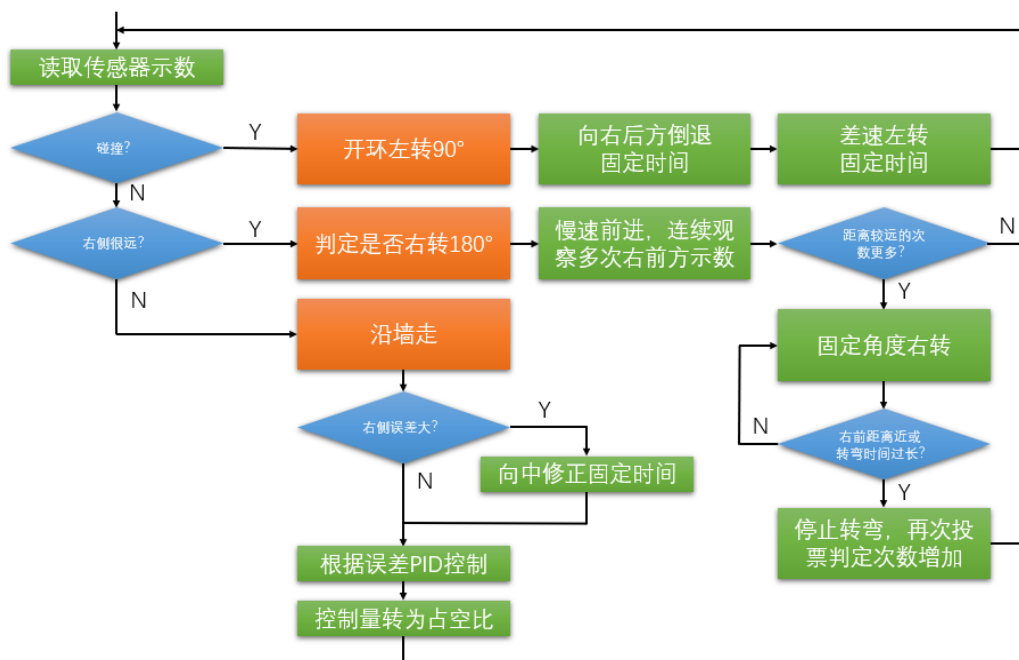


图 2-2 迷宫任务流程图

(1) 右转采样投票算法：当发现超声读数大于一定值时，小车需要做出是否右转的决定。为了避免由于偶然的噪音导致小车做出不恰当的动作，我们自己设计并实现了一种采样平均的算法，实践中发现，该算法能够很好地消除噪声，增强鲁棒性。

具体来说，当小车发现超声读数大于一定值时，会首先将自己的速度减为很慢的直行速度，然后在滑行过程中不断对右前方的红外传感器进行采样，将读数高于阈值的划入“右转”的票，将读数低于阈值的划入“不右转”的票。共计 7 票后，统计是否“右转”的票超过了“不右转”的票，如果是则采取右转直到红外示数小于另一个阈值，当转弯时间超过一个阈值时也会退出，防止过度转弯撞墙。若右转结束了还满足投票条件，则下次投票需要投 25 票，防止连续两次右转。通过这种采样平均的方式，我们成功避免了偶然的测量值影响小车的决策，在实际中达到了很好的效果。

(2) PID 控制贴墙走：在正常情况下，我们设定 50 为超声传感器的基准值。实际的读数于此做差作为误差。在误差范围不同时，我们用了两种控制策略。一是在误差过大时，手动设定转弯值使小车回到正常误差范围内；二是在正常误差范围内，用 PID 控制。为避免转弯过急，转弯幅度有一个特定的最大值。

2.2.4 遇到的问题及解决过程

(1) 两个碰撞传感器同时触发时机器人停机。

对这个问题，我们尝试了多种可能，包括：判断条件为碰撞传感器 A 或碰撞传感器 B 触发；判断条件为只有碰撞传感器 A 触发；判断条件为只有碰撞传感器 B 触发。这里使用的触发判断是 `Button_Read()` 函数。试验结果是，无论代码中涉及几个碰撞传感器的触发，只要两个碰撞传感器都接到芯片上，同时触发就会使得机器人彻底停机，而一时刻触发单个没有任何问题。

我们通过观察 CyPress 板的亮灯情况，推断同时触发会导致整个平台停止工作。但我们至今也未能得知原因所在。最终我们更换了实验板，新的实验板上这个问题就不再出现。

(2) 红外有效测距太短且测距不灵敏。

最初我们测试红外传感器能读取的最大有效距离是 31mm，当时我们并不知道这个范围是不正常的。在最初的迷宫设计中，由于能够测得的距离太短，我们只能把适中距离规定在 22~27mm，这样的距离导致后退左转过程中有很大概率发生碰撞和剐蹭，同时绕墙右后拐弯也十分困难。同时，在测试中我们发现红外传感器测距不灵敏，测量值经常发生跳变。

在更换实验板后，我们发现红外传感器的测距能够达到 42mm 左右，但是在同时接入其他传感器后这一最大距离又会受到影响。后来我们得知红外传感器较容易受线路干扰的影响，因而我们调整了布线，使得红外传感器能够正常稳定工作。

2.2.5 结果与反思

在多次预测试中，机器人在程序指导下能够非常稳定地完成迷宫任务，有极低概率发生略微剐蹭但不影响总体效果。在贴壁直行的过程中，行进路径比较直，与墙壁距离稳定；在右转及右后转弯的过程中，尽管有半径上的抖动，但是未出现剐蹭墙壁的情况；在碰壁左转的过程中，表现流畅良好，后退距离足，转向到位。在过程过渡中，从右后转弯到贴壁的过渡较为顺利，转弯结束时的方向基本与墙壁平行，同时调整较快；贴壁直行过渡到后退左转的过程中，偶有因为直行段过短导致距离调整不到位，造成原地左转剐蹭的情况，但出现的概率极低，且并不会对总体效果有实质性影响。

在验收日，机器人顺利完成了迷宫任务，在验收过程中略微出现了剐蹭墙壁的情况，但对整体效果未有太大影响。

在该任务中，理论上我们可以运用 PID 控制来实现贴壁直行的更科学、效果更优的距离控制，但我们未能及时排除红外传感的硬件问题，导致 PID 实现困难，且大量时间被浪费，因而实际只采用了分段控制方法。这告诫我们要对硬件应有的性能有充分的了解，多与老师和别组同学进行交流，以避免时间和精力上的浪费。

2.3 自选任务

2.3.1 任务描述

考虑扩展项目计划与评估完成可能性，列出三种可能方案的上层设计：

1. 根据提前输入轨迹完成指定动作。
2. 根据传感器反馈在上位机（电脑）勾勒出迷宫地图。
3. 提前录入迷宫信息实现快速寻路算法并最短路径走出迷宫。

综合讨论与初步尝试后决定采用安卓蓝牙操控与电脑交互蓝牙绘制的方案，而且与经典 SLAM 问题比较相似，意义更大。

简单来说任务目标为实现如下功能：手机端通过蓝牙通讯控制小车运动；小车通过红外传感器探知距离，将周围环境信息返回给电脑，绘制一幅周围环境的地图。

2.3.2 设计思路

采用 APPINVENTOR 进行图形化编程制作 APP 实现蓝牙发送控制。

使用前方超声传感器，右侧红外传感器，以及两个碰撞传感器。传感器传来的距离数据，由小车的坐标系转换为世界坐标系后可以确定一个点。这里有一个关键，就是要得到小车的坐标。我们以起始点为零点，由于是开环控制，因此不能够精确地知道小车的坐标，只能由一系列的控制命令推断出其大致位置。这里需要得到小车设置速度（代码里的）与实际速度的映射关系；同时，做转弯实验，得到小车设置转角（代码里的转速差）与转过角度的映射关系。

2.3.4 算法实现

（1）手机操控蓝牙应用的实现

理论依据是使用蓝牙通信连接安卓手机与小车，手机发送控制信号（ASCII 码形式的字符），小车接收且识别后执行相应控制指令。

针对蓝牙传感器模块设计界面：连接与状态按钮，并能显示当前蓝牙状态。四个控制按钮能实现向蓝牙模块发送不同的字母字符（前后左右停止）实现控制。

插入图片使 APP 显得生动，蓝牙部分数据传输比较繁琐，反复尝试 debug 并对缺陷进行了改进，用最稳定的实现方式的字符发送。

编译生成 apk 文件，经测试成功对小车实施了控制。

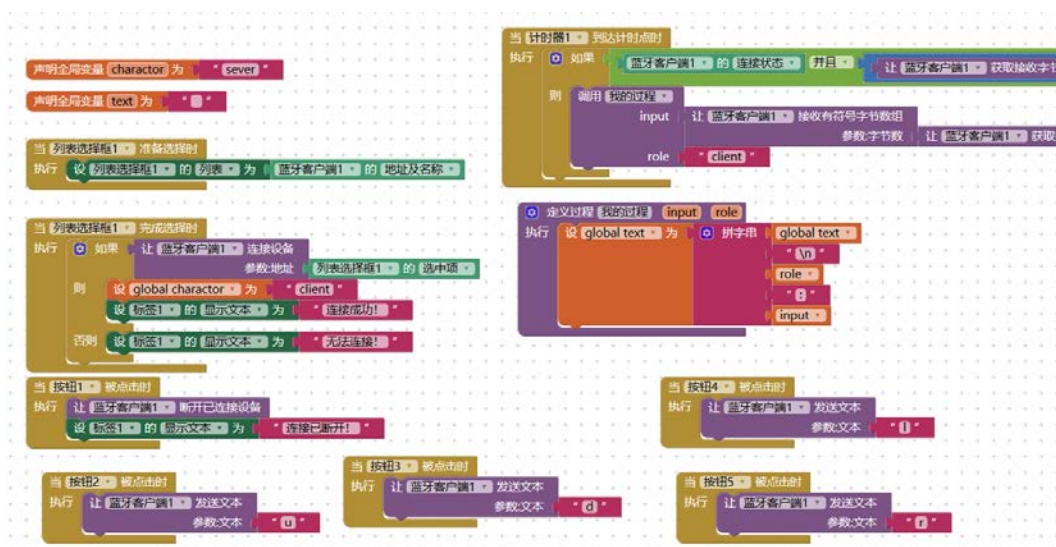


图 2-3 APP Inventor 后台编程界面

(2) 地图绘制的技术细节

小车和电脑的通讯是交互的。电脑端会不断发送运动指令给小车，小车接受指令，同时将超声传感器的信息发送给电脑端，电脑利用先前发送的运动指令信息和获得的超声传感器的信息进行障碍地图的绘制，如此循环往复。

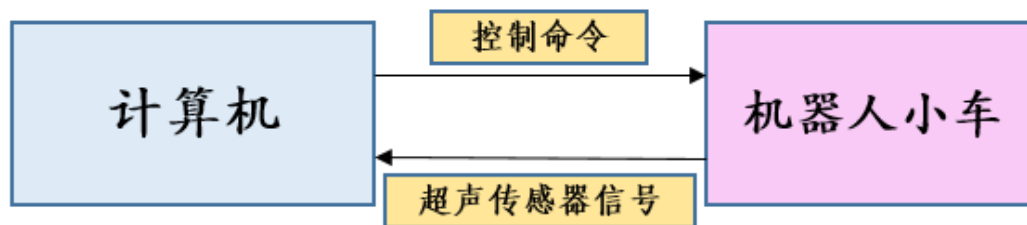


图 2-4 地图绘制信息传输示意

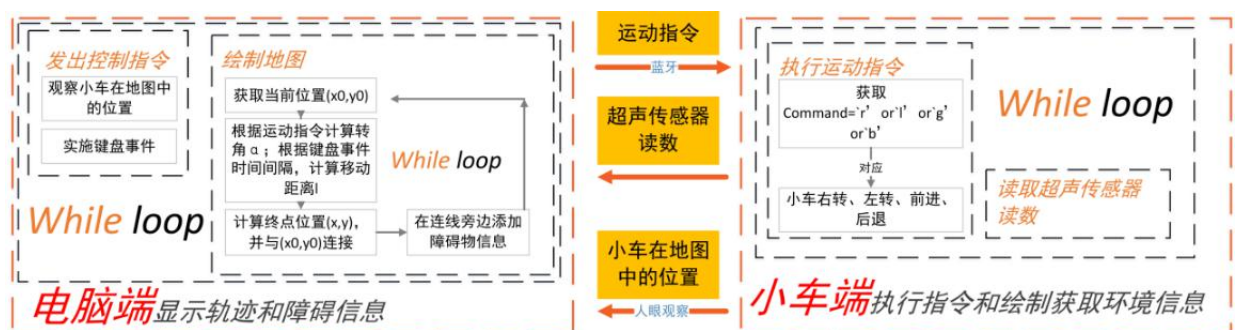


图 2-5 地图绘制具体实现流程

上述过程可以分解成三大部分。

(2.1) 电脑端 python 程序 show.py 获取按键时间，将运动指令发送给小车。

Python 编写的 show.py 程序获得用户的键盘输入，将对应的字符串通过蓝牙发送给小车，小车触发对应的运动模式（上下左右键分别对应直走、后退、左拐、右拐）。

(2.2) 小车获得超声传感器的读数，并且通过蓝牙传给电脑。超声传感器位于小车车身的右侧，其读数在 30 在 1000 之间波动，30 代表距离为 0 或者很近，1000 代表距离为 1m 以上，超声传感器读数和障碍物距离之间大致呈线性关系。由于蓝牙串口只能发送字符串，因此在下位机程序中我们将超声传感器的读数分为 9 段，对应地分别让蓝牙发送 'a','b','c','d','e','f','g','h','i'，python 地图绘制程序得到下位机程序通过蓝牙发过来的字符串，就在小车轨迹旁边的对应位置标注障碍物。

(2.3) 电脑获得小车获超声传感器的读数，并且在图形界面绘制障碍地图。这里要解释两件事情：第一是如何画布上绘制连续轨迹，第二是如何在小车轨迹曲线旁绘制障碍物曲线。第一，如何画布上绘制连续轨迹。画布上所看到的连续轨迹实际上是由一段段小直线组成的。每一段小直线的绘制流程是首先找到小直线的起始点 (x_0, y_0) ，然后确定小直线的长度 l 和偏角 a ，通过 l 和 a 确定小直线的终点 (x, y) ，然后将 (x_0, y_0) 、 (x, y) 进行连接，然后将 (x_0, y_0) 赋值给 (x, y) ，进行下一段小直线的绘制。第二，关于如何在小车轨迹曲线旁绘制障碍物曲线。我们画图程序中，使用在轨迹曲线右侧的同步更新的红色轨迹代表障碍物。红色曲线的绘制也是由小直线连接而成的。不过红色小直线的绘制需要分为直行和转弯两种情况。直行的时候，红色小直线与黑色小直线除了初始位置可能不同外，长度和偏角都相同；转弯的时候，红色小直线的长度除了与黑色小直线长度有关，还与蓝牙读入的超声传感器的数值有对应关系。超声传感器的数值对应着红色和黑色小直线之间的距离 $dist$ ，超声传感器的数值越大，红色

和黑色小直线之间的距离 dist 越大。通过红色小直线和黑色小直线看成共圆心的两段圆弧，我们就能求出小红直线的长度。这样只要我们给出小车轨迹曲线和障碍物曲线的初始点，通过连续不断地计算小黑直线和小红直线，就能把小车轨迹和障碍物轨迹画出来。

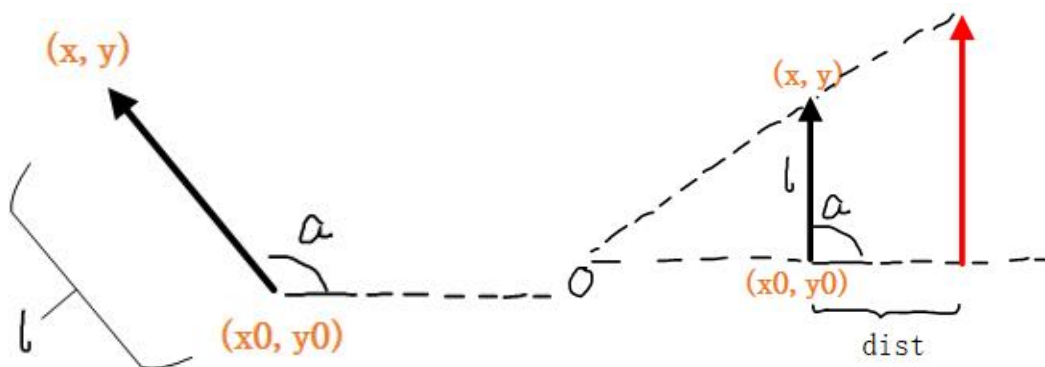


图 2-6 小直线绘制的方法图

2.3.5 遇到的问题及解决过程

(1) 早期的逻辑中，主要问题是电脑端的绘图逻辑和小车运动逻辑不符。

我们反复讨论修改了好几版控制逻辑，最终定下了现在的版本。早期逻辑绘图时当按下键使小车停下时，需要重新按向前键使小车向前，而实际的小车运动在没有收到指令时一直向前（即放开键重新启动）。我们思考之后将逻辑改为当没有键按下时绘图和实际小车都处于静止状态，当按下键时才做运动。事实证明这大大简化了逻辑，也达到了更好的控制效果。

(2) 蓝牙实际调试中的问题

一是当小车电量不足（充电宝五分钟自动断电）时，蓝牙会断开。所以我们采用了在小车不动时也加一个缓慢的电机转速用来消耗电量的方法。二是手动蓝牙连接流程繁琐，需要删除设备-找到新设备-输入密码-连接，并且 Python 和 UART 的串口通信经常失败，为此我在接受端的代码里加了 try 语句和检测语句，增强了鲁棒性。

(3) 开环控制，实际画图和现实的轨迹并不一定重合

解决方案是花费一定的时间手工仔细地调参，尤其是转弯速度和前进速度的参数，使得画图轨迹和实际尽量相似。

2.3.6 结果与反思

我们组在多种方案中选择了最有意义且最有可行性的方案。虽然在实现过程中遇到了许多预计到与未曾预料到的困难，但在大家的通力合作下也一个一个地解决。每个人都收获了许多置石与满足。

任务进行中大家各司其职，妥善地完成自己负责的部分，遇到困难一起讨论钻研。总体进度完全符合预期甚至超前计划，感谢每一位成员所付出的努力。

在实践中我们体会到优化修改是没有尽头的，我们的目标就是做到更好。我们组对小车的绘图程序的简化修改，使得绘图更为流畅和符合逻辑。在后来的优化分析中我们就发现在第一版中程序记录了按键的时间，这其实是没有必要的。

再比如实际迷宫绘制过程中对于图形界面上红色迷宫墙体的显示，修改代码将原来离散点变得更加均匀了一点，并且用 moving average 的算法使得轨迹缓慢追踪新测的点，大大光滑了画出来的曲线。

三 小组成员分工

硬件与巡线：所有人
迷宫：张沛东 陈思哲
自选：牟天昊 忻旻健 叶健龙
摄影摄像视频：陈思哲
报告与 PPT：叶健龙 陈思哲 等
归档：忻旻健 陈思哲 等

四 小组成员个人总结

4.1 张沛东个人总结

本次任务中，我承担了小车的巡线和迷宫任务，同时作为组长督促大家尽快完成每个节点任务，同时我也积累了一些自己的想法。

首先，实际工程问题并不像我们课堂学习到的理论研究那样，并不是模型越高级，理论越复杂，在计算上能够让误差量无限逼近于零的控制算法就是最好的。各个指标的相互权衡，硬件上的误差，现场突发情况等，会让设计时建立的模型与实际有很大的偏差。因此我们首先采用了一个朴素算法作为整体的框架，然后经过调参观察问题，针对出现的问题进行建模和一般化，再设计解决方案。

其次，控制算法的鲁棒性。由于我们一直以为是要求现场考核的方式，所以我们的工作重点更多放在了闭环控制之上，也就是如何才能让我们的小车在哪怕最差的情况下也能走完迷宫全程，因此和其他组的迷宫任务相比，我们组的小车在过弯时并不显得干脆利落，反而拥有更多的迟疑和试探。但这样的迟疑和试探恰恰是要通过设计过程中的持续的裕量考量来实现，也是我们代码中最引以为傲的投票思想，它让我们的小车在无法对自身所处环境形成准确判断时，通过减缓步伐和试探的方式来决定下一步策略，同时考虑到传感器的突变误差，这样的决定是将通过多个连续的时间状态进行投票得到的。因此我们可以自豪地认为，即使我们的小车无法干脆利落，但我们能保证它一定会完成任务。

最后，我们再次深刻意识到传感器的重要性。在任务过程中，多个传感器的不确定性，单个传感器的跳变特性，甚至老化的传感器在使用过程中突然损坏，一度让我的组员现场工作完全无法开展。最终我们更换了全新的传感器，用软件去弥补硬件的不足，才解决了问题。

总之，在这次实践中，我学到了软硬件的结合和小组之间的合作，积累了项目开发的经验，更新了自己传统工程思维，养成专业思维的进步，是在课堂上很难直接获得的，感谢周老师给我们这个实践的机会，也感谢助教的大力帮助。

4.2 陈思哲个人总结

本次任务中，我承担了很多现场调试的工作，形成了自动化控制中一些重要的思维理念。

首先，实际工程问题与理论模型研究，差距很大。并不是说越复杂，理论上无限逼近的控制算法，就是最好的。因为传感器误差，现场突发情况等，会让设计时建立的模型与实际有很大的偏差。实际过程中，一般是一个很朴素的算法，经过大量调参，观察出问题，对此再进一步修正。

其次，控制算法的鲁棒性，是要通过设计过程中的持续的裕量考量来实现。在算法的每一步改进中，都要批判地思考此改进引发的，非期望的情况，以及对应后果的严重程度。在

算法框架基本完成后，鲁棒性的增强则是核心工作。

最后，传感器在工程中极为重要。检测技术课上，我了解到，有一半以上的工业成本，都花在了传感器上，当时十分震惊。但是在任务过程中，多个传感器的不确定性，一度让现场工作完全无法开展。最后我们购买了全新的，高精度的距离传感器，才解决了这个问题。

总之，具体的技能，如硬件配置，软件编写，视频宣传等，都是小的学习与提高。但实践中，通过直观直接的现场工作，养成专业思维的进步，是在课堂上很难直接获得的。

4.3 牟天昊个人总结

在工程实践与科技创新 4E 这门课上，我主要负责自选模块的实验设计和实现。我们小组的自选项目是利用超声传感器和编码器记录小车的运动轨迹并绘制粗略地障碍地图。在整个实验的过程中，我培养了一些重要的思维理念、养成了良好的习惯、掌握了不少软硬件技能，可谓收益匪浅。

在思维理念上，最重要的收获是我第一次真正意识到小组合作的重要性。习惯于独自完成所有任务的我，在与小组其他同学的通力配合的过程中，发现通过与其他同学探讨问题、将问题拆分由不同成员分别解决，配合默契的小组团队能达到 $1+1>2$ 的效果。这次任务中，与忻旻健同学的讨论让我快速理清了图形界面的逻辑设计，张沛东同学对硬件特性的熟知帮助我快速排出 bug，在现场实现预定功能，小组成员对任务的尽职尽责保证了项目的顺利实现。

在习惯上，由于良好的小组氛围中，在其他小组成员的带动下，我一改以往拖拉的毛病，养成了“事情做在 ddl 前面”的习惯，并且越来越享受有计划地安排时间的过程。

在软硬件技能和专业素养上，我熟悉了 python 开发图形界面的流程，了解了蓝牙通讯协议的特点。在实际绘制地图的过程之中，电机转速的不理想、轮子大小的不一致和传感器噪声等误差因素都会使得绘制出来的轨迹与实际轨迹不一致。硬件要求简明的代码逻辑、高的时间和空间效率，逻辑相同的两套实现却可能跑出不一样的结果，在调试的过程中我对硬件逻辑控制思想有了深刻的体会，也有了一些硬件工程师的基本素养。

总而言之，专业技能和素养为日后工作打下基础，而思维理念的改变和良好习惯的养成能影响生活却会影响生活的很多方面，受用终生。感谢王景川老师带来的这门课程！

4.4 忻旻健个人总结

在本次工科创智能小车的课程设计中，我主要协助牟天昊同学一起完成绘制环境地图的任务。除了一些硬件接线和实地测试，我的主要工作偏软件部分，从开始的选题到最后的完工，可以说遇到了很多挑战，但也积累了不少经验。

首先，我觉得程序的简化思维非常重要。在自选模块的主程序上，我们反复讨论修改了好几版控制逻辑。在早期的逻辑中，主要问题是电脑端的绘图逻辑和小车运动逻辑不符。我们思考之后将逻辑改为当没有键按下时绘图和实际小车都处于静止状态，当按下键时才做运动。事实证明这大大简化了逻辑，也达到了更好的控制效果。对于小车的绘图程序，我也在牟天昊同学的第一版上做了简化修改，去掉了计时函数，使得绘图更为流畅。我还对小车走迷宫的程序做了整理，删改了程序中一大堆全局变量，解决了函数调用逻辑混乱的问题，整理好的代码变得清晰，主体逻辑突出。总结这些经验发现，代码中出现 BUG 很多时候来源于程序员自己将逻辑变得复杂了，而很多细节往往是不必要的。所以简化逻辑，抓住主要矛盾，这样的程序往往比较健壮，并且易于解释。

其次，实践中常常会出现理论不讨论的东西，而这些细节会对项目成功与否带来巨大影

响。比如，在实际调试中，蓝牙给了我们很大麻烦。当小车电量不足时，蓝牙会断开，重新手动蓝牙连接流程繁琐。另一个问题是，实际画图和现实的轨迹并不一定重合，需要靠手工仔细地调转弯速度和前进速度的参数，使得画图轨迹和实际尽量相似，这也耗费了我们很多精力。

最后，本次课程设计最大的收获应该是与团队成员合作的经验。在这里特别感谢成员们在各项任务间紧密的配合和负责的态度，让团队的效率得以大幅提高，也让一直有拖延习惯的我学习到了时间规划的重要性。

4.5 叶健龙个人总结

本次实验一如往常的工科创项目一般，初见时一头雾水前路迷茫，进程中拨开迷雾携手并进，迷茫时努力克服与同伴沟通解决问题，最后水落石出成果面前无比的满足感，获得了又一次的成长。主要担任自选项目安卓蓝牙通信的编写，协助进行了绘制地图的调试。

最令人印象深刻的是硬件的调试，直接写代码的情况实在是太过于理想，拿到的小车传感器与车轮转轴都很不尽如人意。极其考验我们的动手能力，安装传感器与和小组同学一起修理卡住的智能车后轮电机令人印象深刻，光是调整好硬件就耗费了一周多的时间。

在项目中我也体会到了团队合作的重要性，大家能力都很强，各司其职完成自己主导的工作。最感谢带给我的改变是同组的成员非常重视时间管理，强调事物在时间表上往前提。这给了我很大的触动，因为我一向喜欢卡着工期结尾交上任务。在组内氛围的督促下我体会到了时间管理的魅力，也慢慢养成了万事往前提的习惯。

在主导的自选项目中用在线 APPINVENTOR 做了安卓开发，以前有过相关经验但是依然面临着许多困难。蓝牙模块部分的语法查阅了很多资料，debug 也有些费神。在这个过程中借助网络资料是非常有用的方法，在调试中也完美实现了安卓蓝牙控制，非常开心。

协助进行地图绘制工作的开展，遇到了许多千奇百怪的困难也提出了自己的看法与建议。大家一起讨论，忻同学对控制代码的理解让我收获了许多。有趣的是代码改进过程中许多备份的删除反而可以更好的提升系统性能，最后结果的实现也需要大量调试参数的过程，感谢同组辛勤的付出！

五 致谢

首先，感谢王景川老师对我们的悉心指导！王老师为我们整理提供了丰富的学习资料，极大方便了我们的学习实践。实验指导文件精炼可靠，为我们的自学过程提供了必要的帮助。王老师在课程的讲授过程中深入浅出，很好地为我们进行了基础知识的铺垫，但又不过于繁琐细碎，让我们有充足的空间进行自主探究和学习。这种气氛有利于我们实践课程的开展。

感谢本课程安排的周报考核制度，具有非常大的现实意义。这一制度形成了一种督促，对成员的“划水”倾向起到了一定的抑制作用，同时“问题”反馈部分也促使我们对问题进行归纳，同时在此过程中试图自己解决。该制度也一定程度上确保了组长能够对整个小组的进度有充分把握。

感谢助教对我们的指导和帮助。在更换实验设备与实验场地铺设的交流过程中，助教都非常热心地帮助我们，还在场提供建议和指导，使我们的排错和更换过程进行得很顺利。最终任务能够顺利完成，一定要感谢助教的帮助。

最后再次感谢 P06 组的成员们的通力合作。在诸多课程的重压下，成员们互助互补，积极交流，团结一致，增强了团队意识也增进了小组成员间的友谊。