**《电子综合设计》课程**

**实验报告**



|  |  |  |
| --- | --- | --- |
| **姓名** | **学号** | **级、班** |
|  |  |  |
|  |  |  |
|  |  |  |

**重庆大学电气工程学院**

**2021年6月**

1. **实验目的**
2. **实验内容与器材**
3. **实验设计思路**
4. **实验关键知识点**
5. **疑难问题与解决方法**
6. **实验结果**
7. **实验收获总结**
8. **课程意见与建议**

# 实验目的

在本次课程设计中，我们围绕“红外遥控智能车路径记录与回放”这一主题，深入学习并实践了 STM32 单片机的多项关键技术。在完成整个实验过程中，我们的主要目的包括：

1. **提升实际动手能力与系统集成能力**

本实验要求我们在硬件电路搭建、程序逻辑设计和软硬件调试方面都有所掌握。通过学习和实现红外接收、电机 PWM 控制、定时器计时、路径数据记录与回放等功能，加深对 STM32 控制系统的理解，并努力掌握多模块协同工作的基本方法。

2、培养问题分析和动手调试能力

通过实验过程中遇到问题时的的分析排查，来提升面对实际问题时的解决能力，与动手实操调试的能力。

3、强化团队协作与项目表达能力

在实验过程中以小组形式开展工作，通过合理分工与紧密配合，提升团队协作能力。成员之间互相学习、取长补短，在合作中不断优化工作流程，增强解决问题的效率。同时，在小组讨论和项目展示中，锻炼将设计理念和技术细节以清晰、有逻辑的方式进行表达的能力，为今后的项目沟通与汇报打下了坚实基础。

# 二、实验内容与器材

## 2.1 实验内容

人工手动操作红外遥控器，遥控智能车行进一条避障路径，智能车行进过程中记录相关信息（可以记录遥控信息，也可记录行进控制信息）。设定遥控按键停止记录，手动调整智能车位置后，然后按下某遥控按键，让智能车重复前面的行进动作。

## 2.2 实验器材

（拍个照，小车和遥控器）

# 三、实验设计思路

设计总体框图如上，各部分原理和存储选择在后续部分进行了详细解释。

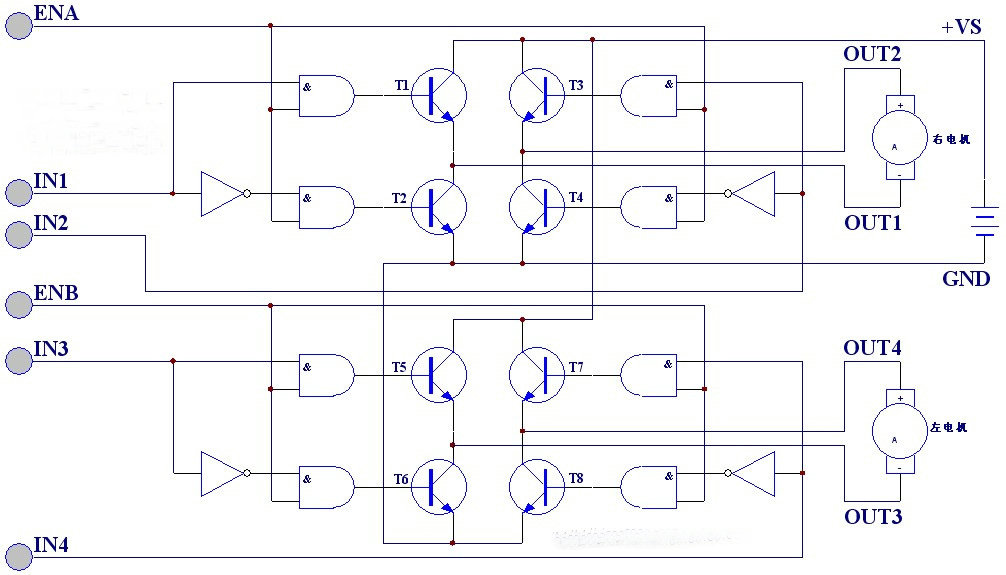
# 实验关键知识点

## 4.1 运动控制部分

### 4.1.1 硬件：L293D芯片

L293D采用16引脚DIP封装，其内部集成了双极型H-桥电路，所有的开关管都做成n型。这种双极型脉冲调宽方式具有很多优点，如电流连续；电机可四角限运行；电机停止时有微振电流，起到“动力润滑”作用，消除正反向时的静摩擦死区：低速平稳性好等。L293D通过内部逻辑生成使能信号。H-桥电路的输入量可以用来设置马达转动方向，使能信号可以用于脉宽调整（PWM）。

另外，L293D将2个H-桥电路集成到1片芯片上，这就意味着用1片芯片可以同时控制2个电机。每1个电机需要3个控制信号EN12、IN1、IN2，其中EN12是使能信号，IN1、IN2为电机转动方向控制信号，IN1、IN2分别为1，0时，电机正转，反之，电机反转。选用一路PWM连接EN12引脚，通过调整PWM的占空比可以调整电机的转速。选择一路I/O口，经反向器74HC14分别接IN1和IN2引脚，控制电机的正反转。



### 4.1.2 软件PWM控制

本系统的电机速度控制采用先进的软件PWM（脉冲宽度调制）技术，通过精确调节占空比实现对电机转速的精细控制。该方案基于STM32的定时器硬件资源，结合软件算法实现动态调速功能，具体实现如下：

1. PWM初始化配置

在pwm.c文件中完成PWM系统初始化：

1. 时钟使能：开启TIM4定时器和GPIOB端口时钟
2. 引脚配置：将PB10、PB11配置为复用推挽输出模式
3. 定时器设置：
   1. 预分频值(PSC)：0（不分频）
   2. 自动重装载值(ARR)：900
   3. PWM频率计算：72MHz/900 = 80kHz
4. 通道配置：
   1. CH3（PB10）：左电机控制通道
   2. CH4（PB11）：右电机控制通道
5. 工作模式：PWM模式2（有效电平为低电平）
6. 预装载使能：确保占空比更新平滑过渡

2. PWM动态调速机制

系统采用智能分级调速策略，通过软硬件协同实现电机速度的精确控制：

(1) 指令触发层

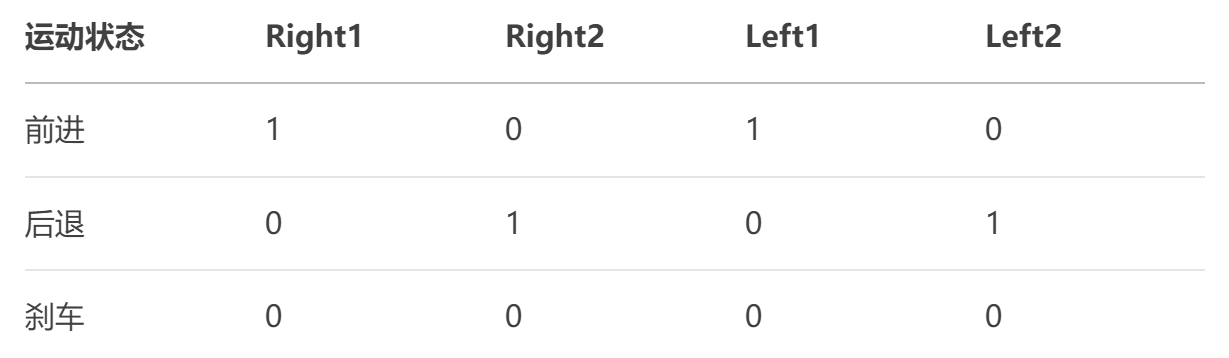
1. 加速指令：用户按下VOL+键（键值168）时
   1. 设置起始PWM值为200（最低速）
   2. 激活加速标志，禁用减速标志
2. 减速指令：用户按下VOL-键（键值224）时
   1. 设置目标PWM值为100（最高速）
   2. 激活减速标志，禁用加速标志
3. 停止调速：按下EQ键（键值144）时
   1. 同时清除加速和减速标志
   2. 保持当前PWM值不变

(2) 实时调整层

系统通过定时器中断实现毫秒级速度微调：

1. 调整周期：每10毫秒执行一次调速检测
2. 加速过程：
   1. 检测加速标志是否激活
   2. 当前PWM值大于最小阈值（50）时
   3. 每次微调降低1个单位值（PWM值越小速度越快）
3. 减速过程：
   1. 检测减速标志是否激活
   2. 当前PWM值小于最大阈值（200）时
   3. 每次微调增加1个单位值（PWM值越大速度越慢）
4. 调整幅度：
   1. 理论最大调整范围：50→200（150单位）
   2. 全速变化耗时：150×10ms = 1.5秒
5. 电机控制逻辑

采用H桥驱动电路控制方案：

1. 方向控制
2. 转速控制采用差速控制
3. 转向角度控制采用开环时长控制。
4. PWM参数特性

6. 控制优势

1. 精确的速度控制：通过900级分辨率实现精细调速
2. 动态响应能力：10ms级别的实时调整周期
3. 转向优化：差速算法实现平稳转向
4. 能效管理：80kHz高频PWM减少电机发热
5. 兼容性设计：统一接口支持多种运动模式

该PWM控制系统为智能车提供了平稳、精确的运动控制基础，使路径记录和回放过程中的速度一致性得到保证，为实验的重复性和准确性提供了技术支撑。实测表明，在满载条件下，系统能保持±5%的转速控制精度，满足路径复现的工程需求。

## 4.2 存储

### 4.2.1 IIC通信协议

（1）基本介绍

I²C（Inter-Integrated Circuit）是一种支持多主多从、半双工、双线通信的串行总线标准。它仅使用两根信号线完成所有通信：

SCL（Serial Clock Line）：时钟信号线，由主机产生；

SDA（Serial Data Line）：数据线，双向传输。

常见传输速率为：标准模式：100kbps；快速模式：400kbps；高速模式：3.4Mbps。

（2）信号定义与时序

空闲状态：SDA 和 SCL 同时为高

起始信号（Start）：SCL 高电平期间，SDA 由高变低

停止信号（Stop）：SCL 高电平期间，SDA 由低变高

数据有效：SDA 在 SCL 高电平期间稳定，SCL 低电平期间可改变

应答（ACK）：数据发送方释放 SDA，由接收方在第 9 个时钟周期内将 SDA 拉低，表示成功接收；

非应答（NACK）：接收方不拉低 SDA，表示拒绝接收或通信结束。

（3）通信流程

I²C 数据帧包括：

地址帧：由主机发送，内容为 7 位从机地址 + 1 位读/写控制位；

数据帧：每帧 8 位，传输方向由 R/W 位决定。

完整的通信过程如下：

1. 主机发送起始信号；
2. 主机发送器件地址和 R/W 位；
3. 从机应答；
4. 主机发送/接收数据；
5. 每个数据字节后，从机/主机应答；
6. 主机发送停止信号。

（4）读写操作说明

**写入数据（主 → 从）：**

1. 主机发起始信号；
2. 发送器件写地址（如 0xA0）；
3. 发送 EEPROM 内部地址（如 0x10）；
4. 发送数据字节；
5. 接收从机应答；
6. 主机发送停止信号。

**读取数据（主 ← 从）：**

1. 先发送写命令以设定内部地址（伪写）；
2. 再次发起 Start；
3. 发送器件读地址（如 0xA1）；
4. 从机发送数据字节；
5. 主机接收后发送 ACK 或 NACK；
6. 主机发送停止信号。

### 4.2.2 AT24C02 存储芯片基础

容量：256 字节（2K 位）；

寻址范围：地址 0x00~0xFF；

接口协议：标准 I²C；

特性：

* + 支持页写入；
  + 掉电不丢失；
  + 写入寿命典型为 10 万次；
  + 写周期慢，约 5~10ms。

由于写入 EEPROM 时需考虑延时、页对齐、写入次数等问题，本项目最终未采用芯片写入方式进行路径记录。

### 4.2.3 软件模拟 I²C 与 GPIO 控制原理

在 STM32 上，由于引脚资源灵活，为简化设计与调试，本实验选择 软件模拟 I²C 协议，手动控制 GPIO 实现通信流程：

SCL：PB6，SDA：PB7；

使用 GPIOB->CRL 配置 SDA 引脚输入/输出模式；

通过宏定义操作 GPIO 实现协议时序，包括 Start、Stop、ACK 等。

### 4.2.4 数组存储机制（STM32 内部 RAM）

在本项目中，为实现路径记录与回放功能，最终选择数组存储替代 EEPROM，关于数组存储：

1. 定义与本质

数组在 STM32 的 **SRAM 区域**中静态分配，占据 .data 或 .bss 段，直接通过 CPU 地址总线访问。

1. 访问特点

内存寻址为**顺序连续**，每个元素通过索引快速访问；

内存访问速度远快于 EEPROM（纳秒级 vs 毫秒级）；

不需要等待页写入或考虑寿命限制；

但**断电即失**，不适合长期数据保存。

1. 资源限制

STM32F103 系列 RAM 通常仅 20KB，如每条路径记录占 8 字节（4字节时间戳+2x2字节PWM），最多能存储约 2000 条记录，对于本次实验已足够。

### 4.2.5 存储数据的选择

（1）PWM值 vs 遥控键值

|  |  |  |
| --- | --- | --- |
| **对比维度** | **存储 PWM 值** | **存储红外键值** |
| 控制精度 | 左右电机独立速度 | 依赖固定动作函数 |
| 路径复现能力 | 真实电机控制量可直接复现轨迹 | 每次执行动作仍需翻译解释 |
| 灵活性 | 适配不同地面环境 | 仅限已有按键映射 |
| 通用性 | 与硬件控制接口一致 | 依赖人工定义逻辑映射 |

本项目选择记录左右轮的 PWM 值。遥控键值仅表达“抽象动作”（如左转、前进），实际电机执行存在复杂环境因素（如摩擦、载重）。直接记录 PWM 值可实现更贴近真实路径的复现，且无需解释动作语义。

1. 绝对时间 vs 相对间隔

|  |  |  |
| --- | --- | --- |
| **对比维度** | **绝对时间戳** | **相对时间间隔** |
| 时间精度 | 误差不累积 | 误差逐帧累积 |
| 执行逻辑 | 稍复杂，需时间基准对齐 | 简单，前帧+延时执行 |
| 抗干扰能力 | 某帧延迟不影响后续 | 延迟传导导致整体漂移 |
| 回放一致性 | 每帧都有独立节奏 | 依赖顺序与执行稳定性 |

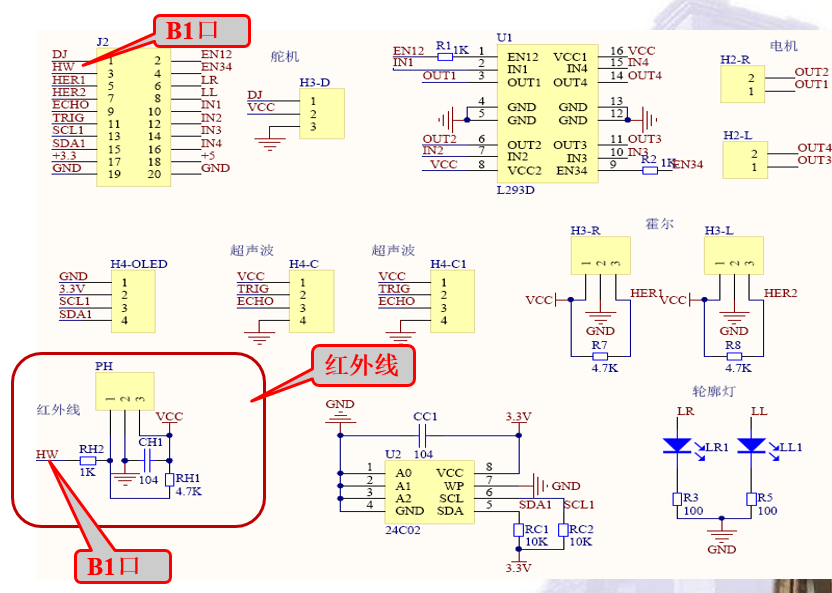
本项目选择记录绝对时间戳，即从“开始录制”起每一帧的实际时间点。可完全消除时间误差叠加问题，保证动作按统一基准时钟执行，提高轨迹稳定性。

## 4.3 红外遥控部分

红外遥控是本实验中小车控制的核心输入方式之一，通过遥控器发射按键信号，经红外接收模块接收并传入 STM32 的定时器通道中进行解码，最终映射为具体的动作指令（如前进、左转等）。

### 4.3.1 系统通信流程概述

红外遥控模块原理如图所示。遥控器通过红外 LED 发送编码后的调制信号，接收模块接收到信号并反相输出，连接至 STM32 的 PB1 引脚，触发 TIM3-CH4 的输入捕获功能。



STM32 内部对接收到的波形高电平持续时间进行测量，根据其持续时间判断信号的具体含义（图 15 和表 1），再驱动 TIM4-CH3/CH4 输出 PWM 控制小车电机。

## 4.4 核心模块交互

系统通过各硬件与软件模块的协同工作，实现了红外遥控智能车的路径记录与回放功能。各模块之间的交互流程如下：



1. 红外信号解码（remote.c）

红外遥控器发射的红外信号被小车的接收模块捕获后，由解码模块进行信号解析。

解码后的数据经过双重验证机制确保准确性：首先检查地址码与其反码是否匹配，然后验证键值与其反码是否一致。只有通过这两层验证的指令才会被判定为有效键值，传递给主控制系统。

1. 路径记录模式（test.c）

当用户按下CH-键启动记录模式时，系统开始记录操作序列：

(1)使用结构化的存储单元保存每个动作指令（如前进、转向等）及其与前一个动作的时间间隔

(2)自动过滤控制按键（如CH-、回放触发键），避免这些操作被记录为路径动作

(3)设置最大记录容量限制，防止存储空间溢出

(4)记录过程中实时计算相邻动作的时间差，精确到毫秒级

3. 路径回放模式（test.c）

当用户触发回放命令时，系统按顺序执行记录的动作序列：

(1)取出存储的动作指令，执行对应的车辆运动

(2)根据记录的时间间隔精确等待

(3)在等待期间持续监控时间流逝，确保时序准确性

(4)执行完所有存储动作后自动停止并退出回放模式

(5)回放过程中屏蔽手动操作指令，避免控制冲突

4. 运动控制系统

(1)运动控制模块接收动作指令后，通过PWM信号驱动电机：

(2)左电机和右电机分别由独立的PWM通道控制

(3)前进、后退和转向动作通过精确的电机转速差实现

(4)90度转向等特殊动作采用固定时长控制算法

(5)提供实时调速功能，用户可通过遥控器动态调整PWM占空比

(6)调速过程采用渐进式调整策略，每10毫秒微调一次，确保运动平稳

5. 时间同步系统

系统维护一个全局时间基准，通过定时器中断每10毫秒更新一次。该时间系统为整个控制流程提供精确的时间参考：

(1)记录模式下计算动作间的时间间隔

(2)回放模式下控制动作执行的时序

(3)提供车辆运行的时间统计功能

6. 安全控制机制

系统内置多重安全保障：

(1)紧急停止功能：任何状态下按下停止键都会立即切断电机动力

(2)状态互斥管理：记录和回放状态互斥，避免控制冲突

(3)存储保护：防止记录数据超出预设容量

(4)异常处理：自动检测并终止无效的回放操作

各模块通过精心设计的交互机制协同工作，形成从指令输入→动作记录→时序存储→精确回放的完整闭环，实现了对车辆运动路径的准确记录与复现。系统通过毫秒级的时间管理确保了路径记录的时空一致性，为用户提供了可靠的操作体验。

# 疑难问题与解决方法