XXXXX报告

班级：XXXXXXXX

学号：XXXXXXXXXX

姓名：XXX

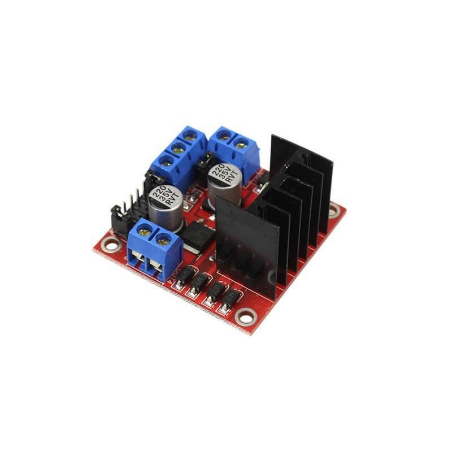
1. 简介

四路红外线循迹小车，由STC89C52RC单片机驱动，采用5V稳压锂电池供电。此车循迹功能稳定，容错性强，在特定轨道模板下可应对复杂环境。

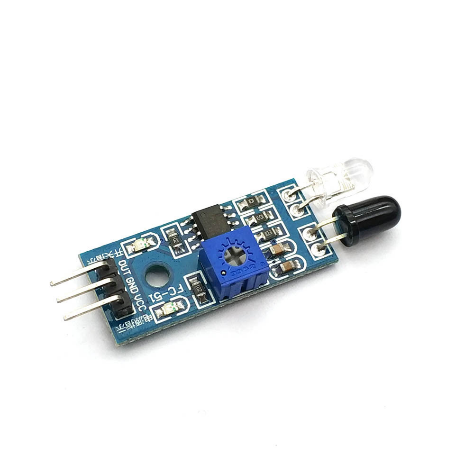
1. 硬件配置
2. STC89C52RC单片机

STC公司生产的一种低功耗、高性能CMOS8位微控制器，具有8K字节系统可编程Flash存储器。STC89C52使用经典的MCS-51内核，但是做了很多的改进使得芯片具有传统51单片机不具备的功能。在单芯片上，拥有灵巧的8 位CPU 和在系统可编程Flash，使得STC89C52为众多嵌入式控制应用系统提供高灵活、超有效的解决方案。

1. L298N电机驱动模块

使用ST公司的L298N作为主驱动芯片，具有驱动能力强，发热量低，抗干扰能力强的特点。该芯片采用15脚封装。主要特点是：工作电压高，工作电压高 可达46V；输出电流大，瞬间峰值电流可达3A，持续工作电流为2A；额定功率25W。内含两个H桥的高电压大电流全桥式驱动器，可以用来驱动直流电动机和步进电动机、继电器线圈等感性负载;采用标准逻辑电平信号控制；具有两个使能控制端，在不受输入信号影响的情况下允许或禁止器件工作有一个逻辑电源输入端，使内部逻辑电路部分在低电压下工作；可以外接检测电阻，将变化量反馈给控制电路。该芯片可以驱动一台两相步进电机或四相步进电机，也可以驱动两台直流电机。模块可以使用内置的78MO5通过驱动电源部分取电工作，但是为了避免稳压芯片损坏，当使用大于12V驱动电压的时候，请使用外置的5V逻辑供电。模块使用大容量滤波电容，续流保护二极管，可以提高可靠性。

1. 红外线循迹模块

该传感器模块对环境光线适应能力强，其具有一对红外线发射与接收管，发射管发射出一定频率的红外线，当检测方向遇到障碍物（反射面）时，红外线反射回来被接收管接收，经过比较器电路处理之后，绿色指示灯会亮起，同时信号输出接口输出数字信号（一个低电平信号），可通过电位器旋钮调节检测距离，有效距离范围2-30cm，工作电压为3.3V-5V。该传感器的探测距离可以通过电位器调节、具有干扰小、便于装配、使用方便等特点，可以广泛应用于机器人避障、避障小车、流水线计数及黑白线循迹等众多场合。

1. 5V定制稳压电池
2. 红外线遥控模块
3. 有源蜂鸣器
4. 最小系统板
5. 车模、杜邦线等其他辅助配件
6. 软件配置

软件驱动采用全自主研制算法，循迹功能拥有超强的容错性和稳定性。算法可以实现黑线循迹、发车鸣笛、停车鸣笛、识别停车区、倒车入库等。

此算法默认使用Keil C51进行编译。若使用MIT开源项目keil2sdcc对文件进行语法转换，可在PlatformIO平台上使用捆绑SDCC编译器进行编译，同时配合stc-isp软件烧录。

1. 特色功能

算法已在GitHub开源（详细地址见下一节），可根据实际情况对各类小车进行适配。同时可以适配Visual Studio Code + PlatformIO IDE、Visual Studio Code + Embedded IDE、Clion + PlatformIO IDE、Keil C51等多种开发环境环境进行Windows、MacOS以及Linux下的全平台开发。

特殊红外遥控算法，可以实现无接触发车，并可以后期配套搭载红外遥控算法远程遥控行驶（默认只能实现全自动循迹行驶），且兼容所有红外遥控器。

使用定制5V稳压锂电池供电，不会出现电涌，电压符合各个模块的供电最高标准，可以使各个模块全功率运行，保障小车的正常使用。

1. 代码示例

由于报告版面有限，且此项目已在GitHub开源，项目地址：<https://github.com/HosonLi/InfraredTrackingCar.git> 或搜索：InfraredTrackingCar.

主要算法示例代码：

1. **while** (1)
2. {
3. **if** (P1\_7 == 0)
4. {
5. P1\_6 = 0;
6. **break**;
7. }
8. }
9. CarForward();
10. **while** (1)
11. {
12. **if** (I1==1 && I2==1 && I3==1 && I4==1)
13. {
14. P1\_6 = 1;
15. Delay(70);
16. **break**;
17. }
19. }
20. **while** (1)
21. {
22. **if** (I1==0 && I2==0 && I3==0 && I4==0)
23. {
24. CarForward();
25. }
27. **if** ((I1==0 && I2==1 && I3==0 && I4==0) || (I1==1 && I2==0 && I3==0 && I4==0))  //小车偏左，小车向右移动
28. {
29. CarRightStop();
30. }
32. **if** ((I1==0 && I2==0 && I3==1 && I4==0) || (I1==0 && I2==0 && I3==0 && I4==1))  //小车偏右，小车向左移动
33. {
34. CarLeftStop();
35. }
37. **if** ((I1==0 && I2==0 && I3==1 && I4==1) || (I1==0 && I2==1 && I3==1 && I4==1)) //直角左拐
38. {
39. CarForward();
40. Delay(180);
41. CarLeftStop();
42. **while** (1)
43. {
44. **if** (I3 == 1)
45. {
46. Delay(10);
47. **break**;
48. }
49. }
50. Line++;
51. }
53. **if** ((I1==1 && I2==1 && I3==0 && I4==0) || (I1==1 && I2==1 && I3==1 && I4==0)) //直角右拐
54. {
55. CarForward();
56. Delay(180);
57. CarRightStop();
58. **while** (1)
59. {
60. **if** (I2 == 1)
61. {
62. Delay(10);
63. **break**;
64. }
65. }
66. Line++;
67. }
69. **if** (Line == 4)
70. {
71. **break**;
72. }
73. }
75. **while** (1)
76. {
77. **if** (I1==0 && I2==0 && I3==0 && I4==0)
78. {
79. CarForward();
80. }
82. **if** ((I1==0 && I2==1 && I3==0 && I4==0) || (I1==1 && I2==0 && I3==0 && I4==0))  //小车偏左，小车向右移动
83. {
84. CarRightStop();
85. }
87. **if** ((I1==0 && I2==0 && I3==1 && I4==0) || (I1==0 && I2==0 && I3==0 && I4==1))  //小车偏右，小车向左移动
88. {
89. CarLeftStop();
90. }
92. **if** ((I1==1 && I2==1 && I3==1 && I4==1) || (I1==0 && I2==0 && I3==1 && I4==1) || (I1==0 && I2==1 && I3==1 && I4==1) || (I1==1 && I2==1 && I3==0 && I4==0) || (I1==1 && I2==1 && I3==1 && I4==0))
93. {
94. unsigned **int** a = 0;
95. CarForward();
96. Delay(180);
97. CarRightStop();
98. Delay(800);
99. **while** (1)
100. {
101. **if** (I2 == 1)
102. {
103. Delay(10);
104. **break**;
105. }
106. }
107. **break**;
108. }
109. }
111. **while** (1)
112. {
113. **if** (I1 == 0 && I2 == 0 && I3 == 0 && I4 == 0)
114. {
115. CarBackward();
116. }
118. **if** ((I1==1 && I2==1 && I3==1 && I4==1) || (I1==0 && I2==0 && I3==1 && I4==1) || (I1==0 && I2==1 && I3==1 && I4==1) || (I1==1 && I2==1 && I3==0 && I4==0) || (I1==1 && I2==1 && I3==1 && I4==0))
119. {
120. P1\_6 = 0;
121. **break**;
122. }
123. }
124. CarStop();
125. Delay(10);
126. P1\_6 = 1;
127. **while** (1);
128. 总结

由于本人能力有限，如有错误或更好地优化方法，欢迎指出！