

文章编号: 2095-2163(2019)04-0149-06

中图分类号: TP23

文献标志码: A

# 基于嵌入式智能小车控制系统设计

周 恒, 王民慧

(贵州大学 电气工程学院, 贵阳 550025)

**摘 要:** 智能汽车是当今车辆工程领域研究的前沿,体现了车辆工程、人工智能、自动控制、计算机等多个学科领域理论技术的交叉和综合,是未来汽车发展的趋势。本系统采用 S9KEAZ128AMLK 芯片为核心处理器,对智能小车的转向及动力系统进行设计,通过 ESP8266WiFi 模块,将智能小车各个传感器的数据通过简易的通信协议实时传输给本设计中的上位机,以达到智能小车数据的实时监控及控制。

**关键词:** S9KEAZ128AMLK; ESP8266; 智能小车

## Design of intelligent vehicle control system based on embedded system

ZHOU Heng, WANG Minhui

(The Electrical Engineering College, Guizhou University, Guiyang 550025, China)

**[Abstract]** Intelligent vehicle is the frontier of vehicle engineering research. It reflects the intersection and synthesis of theory and technology in vehicle engineering, artificial intelligence, automatic control, computer and other disciplines. It is the trend of future automobile development. The S9KEAZ128AMLK chip is the core processor of the system. The steering and power system of the intelligent car is designed. Through ESP8266 WiFi module, the data of each sensor of the intelligent car is transmitted to the upper computer in real time through simple communication protocol, so as to achieve real-time monitoring and control of the data of the intelligent car.

**[Key words]** S9KEAZ128AMLK; ESP8266; intelligent car

## 0 引言

智能汽车是当今车辆工程领域研究的前沿,体现了车辆工程、人工智能、自动控制。为了使智能汽车获得更快的速度、更好的控制性能,智能小车上位机的研究也成为主要研究方向。随着无线通信以及上位机的概念逐渐被众人所熟知,设计一个完整的上位机调试平台对于智能小车来说也成了必不可少的事情。本设计以 S9KEAZ128AMLK 芯片为核心处理器设计,下位机控制系统以 LabVIEW 为平台设计上位机控制系统。当智能小车开始寻迹时,上位机控制系统用于对智能小车寻迹过程中数据的采集与监控,这样便可以知道智能小车的实时运动状态,通过小车的实时运动状态使用者可以调整出最优 PID 参数,使小车获得更快的速度。

## 1 智能小车硬件设计

### 1.1 智能小车系统架构

智能小车控制系统以 S9KEAZ128AMLK 芯片为核心处理器,通过 LC 选频电路对赛道信号进行

检测之后通过信号放大电路将 LC 选频电路采集的信号放大之后传到单片机的 AD 口,将其转换成单片机能够处理的数字信号发送到单片机处理。KEAZ128 单片机将电感采集的信号、编码器采集的信号以及 MPU6050 陀螺仪加速度计采集的信号通过均值滤波,然后通过不同的算法对其获取一个数值,将该数值发送到 PWM 脉宽调制控制电机输出<sup>[1]</sup>。在调试智能小车时人们将数据通过 WiFi 模块发送本设计的上位机进行观察,同时为了方便也将数据发送到液晶显示屏显示。此外 WiFi 模块也接收上位机发送过来的信号,并将其信号发送到单片机处理(如一键停车等)。硬件系统结构如图 1 所示。

### 1.2 智能小车硬件介绍

#### 1.2.1 LC 信号选频电路设计

根据麦克斯韦电磁场理论,交变电流会在其周围产生交变的电磁场。在赛道上载流导线内通过的交流电流频率约为 20 kHz,电流大小约为 100 mA,其产生的电磁波属于甚低频(VLF)电磁波<sup>[1]</sup>。甚低频频率范围处于工频和低频之间,为 3~30 kHz,波长为 100~10 KM。

**作者简介:** 周 恒(1996-),男,硕士研究生,主要研究方向:嵌入式;王民慧(1962-),女,学士,副教授,主要研究方向:嵌入式系统、自动化装置。

**通讯作者:** 王民慧 Email:468427133@qq.com

**收稿日期:** 2019-04-05

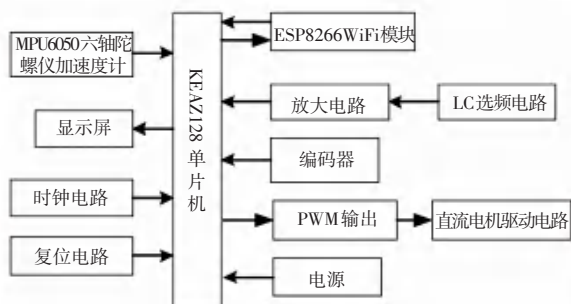


图1 智能小车硬件系统框图

Fig. 1 Smart car hardware system block diagram

根据电磁学的知识,载流导线周围充满交变的磁场,在当中放置一个电感线圈,电磁感应会使线圈内产生感应电动势,该感应电动势信号比较弱,感应电压最大只能达到百毫伏级,此外空间中还存在很多频率、强弱均不同的磁场,如果直接采集,采集的信号噪声会比较大。所以针对赛道上的20 kHz交流磁场采用LC选频电路对信号进行选频,这样便可滤除噪声的干扰<sup>[9]</sup>。LC谐振电路如图2所示。

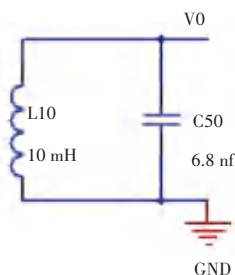


图2 谐振电路

Fig. 2 Resonant circuit

根据上述电路,  $C$  为并联的校正电容,  $L$  为感应线圈的电感量,电路谐振频率的公式为:

$$f = \frac{1}{2\pi\sqrt{LC}} \quad [1]$$

已知:  $f = 20$  kHz,  $L = 10$  mH,可以计算出谐振电容的容量为:

$$C = \frac{1}{(2\pi f)^2 L} = 6.33 \times 10^{-9} (F) \quad [1].$$

因目前只有与其接近的6.8 nF的电容,因此谐振电路中的校正电容的容量为6.8 nF。经实验验证,选用10 mH的电感和6.8 nF的校正电容可以很好地获取赛道信号。

### 1.2.2 信号放大电路的设计

由于LC谐振电路得到的信号是交变的电压信号,且电压幅值只有百毫伏,经A/D转换后数值差异并不明显。因此,还需要将采集的信号进一步放大<sup>[4]</sup>。OPA2350运放芯片输出类型为:满摆幅,理论供电电压为5 V,可以将采集的微小的电压信号

放大到5 V,放大后的电压即可达到供电电压的幅值,其压摆率可达22 V/us,增益带宽积可达33 MHz,很好地满足放大需求。

通过Multisim软件仿真观察到输入信号很明显地按照本设计放大,可以证明本次电路设计是正确的。仿真结果如图3所示。

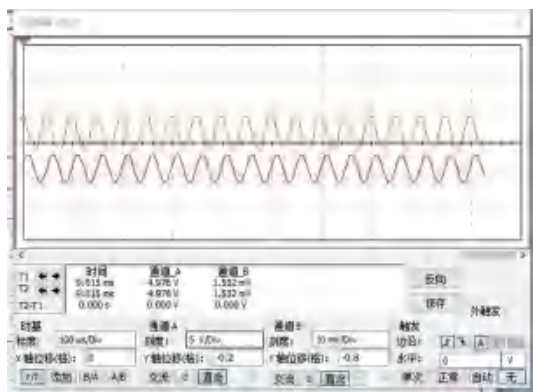


图3 Multisim软件仿真结果

Fig. 3 Multisim software simulation results

### 1.2.3 检波电路设计

由上文可知,OPA2350芯片所设计的放大电路只是将信号放大了,实际上还是交变的电压信号。如果将交变的电压信号直接给单片机的A/D口,无疑会大大增加单片机对信号滤波的处理难度。因此,在电路设计中还需要一个检波电路,将交变的电压信号整流成直流信号送到单片机的A/D口<sup>[7]</sup>。该电路采用倍压整流原理设计,检波电路如图4所示。

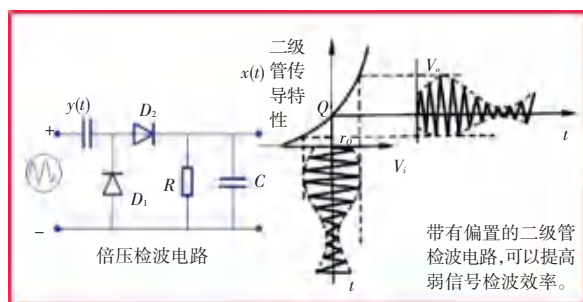


图4 倍压整流电路

Fig. 4 Voltage doubler rectifier circuit

通过Multisim软件仿真验证该设计的检波电路是正确的,其仿真结果如图5所示。

### 1.2.4 稳压电路设计

智能小车车模使用7.2 V 2 000 mAh Ni-cd 供电,而ESP8266 WiFi模块、陀螺仪、加速度传感器和液晶显示屏均使用的是3.3 V的电源。编码器、电磁传感器模块以及单片机最小系统需要5 V电源。因此电路设计中必须将7.2 V的电源电压降至5 V以及3.3 V的供电电源。本设计通过德州仪器公司

生产的 TPS 系列稳压芯片,对电池电压进行降压至 3.3 V 和 5 V,因为该芯片稳压后会有较大的纹波,因此设计电路时采用 0.1  $\mu\text{f}$  的瓷片电容滤除高频信号、10 $\mu\text{f}$  的电解电容滤除低频信号进行滤波处理,得到一个稳定的直流电源。

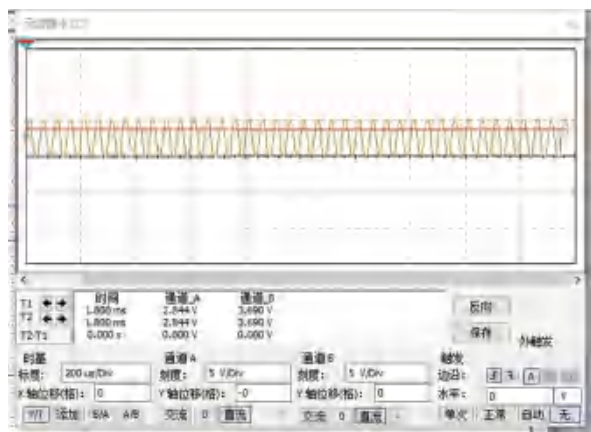


图 5 倍压检波电路仿真结果

Fig. 5 Double voltage detection circuit simulation results

### 1.2.5 电机驱动电路设计

智能小车在寻迹过程中电机的速度是根据编码器反馈值进行实时调速的。因此,在设计时不能直接将电源对电机进行供电,本设计采用脉宽调制实现了对电机速度的控制。S9KEAZ128AMLK 单片机在接受并处理完传感器传回赛道的信息后,得到一个目标速度,然后通过脉宽调制信号来控制电机的转速。由于 S9KEAZ128AMLK 单片机输出信号的电流较小,不能驱动电动机转动,所以需要设计一个驱动模块来放大单片机的输出信号,实现对电机的驱动<sup>[3]</sup>。因此,电机驱动电路中使用 74HC244 芯片对 PWM 信号进行缓冲,采用 IR2104SMOS 驱动管和 LR7843MOS 管设计了如图 6 所示的电机驱动电路。

因 IR2104SMOS 驱动管需要接入 12V 或者更高的电压才可以使得 MOS 管导通,因此升压电路采用 MC34063 芯片,根据其数据手册,设计了 MC34063 升压电路。

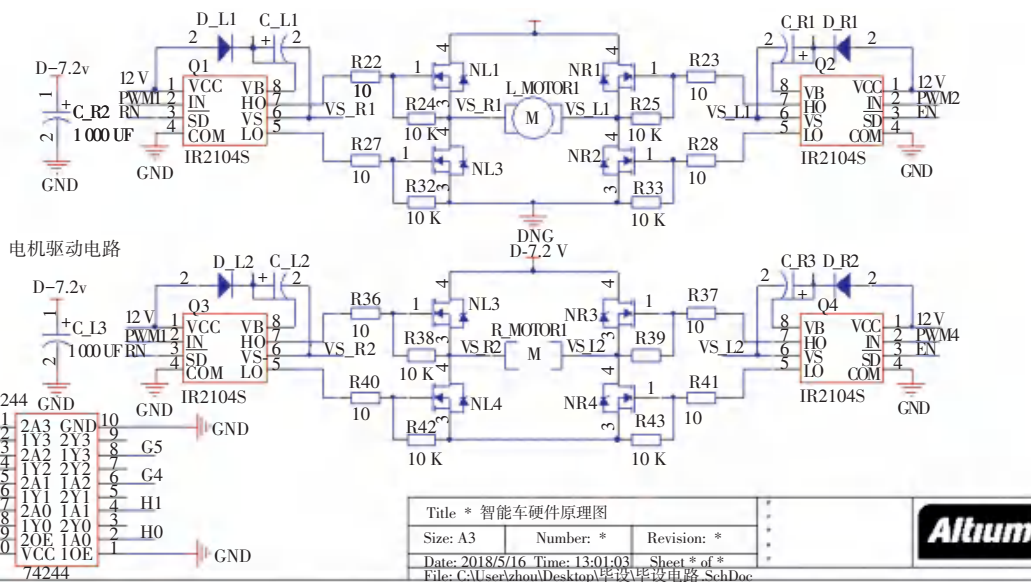


图 6 电机驱动电路

Fig. 6 Motor drive circuit

## 2 软件设计

### 2.1 下位机软件系统设计

#### 2.1.1 主程序设计

由于智能小车软件控制系统设计较为复杂,所以采用模块化的思想来实现该控制系统的设计。在主程序中主要对 S9KEAZ128AMLK 单片机、MPU6050 的初始化以及 ESP8266WiFi 模块的初始化,保证各个模块能够正常运行,然后打开中断,允许中断响应去执行相应的子程序。

#### 2.1.2 中断子程序设计

S9KEAZ128AMLK 单片机通过定时器 0 中断去执行各个子任务。当 Pit\_count 计数值为 1 时 S9KEAZ128AMLK 单片机对传感器所获得的信号进行 A/D 采集,并将该信号进行均值滤波后调用信号处理函数对信号进行处理;当 Pit\_count 计数值为 2 时,对所获得的 MPU6050 传感器数据进行互补滤波得到小车的准确姿态,并调用 BalanceCtrl() 函数对小车的平衡进行控制;当 Pit\_count 计数值为 3 时,通过调用 Get\_Ddy() 函数得到小车前进路线时偏离

轨道的偏差,并通过 DirecontrolPID() 函数对小车的行驶路线进行校正使得小车能够保持正确的路线行驶;当 Pit\_count 计数值为 4 时,对 Pit\_count1 的值进行加 1 并判断 Pit\_count1 的值是否已累加至 20,如果值为 20 则调用 Get\_Speed() 函数获取小车的速度值,并调用 MakeSpeed 函数通过增量式 PID 算法对小车的速度进行控制,然后将 Pit\_count1 计数值清零;当 Pit\_count 计数值为 5 时将 Pit\_count 计数值清零并返回主程序。中断子程序流程如图 7 所示。

### 2.1.3 获取小车姿态算法设计

若从陀螺仪角速度获取角度信息需经过积分计算,当角速度信息存在微小的偏差或漂移时;而经过积分运算后得到的角度信息就会因为累计误差而无法得到小车准确的姿态信息;因此本设计利用加速度计所获得角度信息与陀螺仪积分后的角度进行比较,将比较后的误差经过比例  $1/T_z$  放大之后与陀螺仪输出的角度信号叠加之后再行积分,经过比例、积分环节后得到小车准确的姿态信息。其算法过程如图 8 所示。

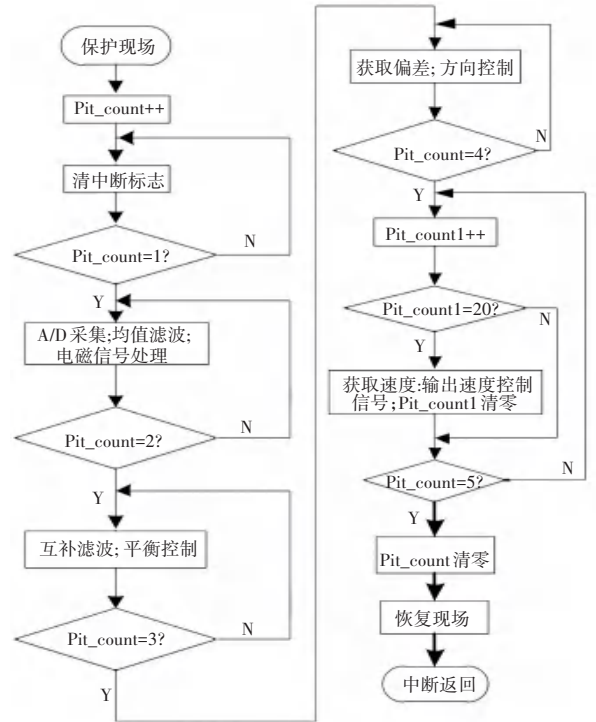


图 7 中断子程序流程图

Fig. 7 Interrupt subroutine flow chart

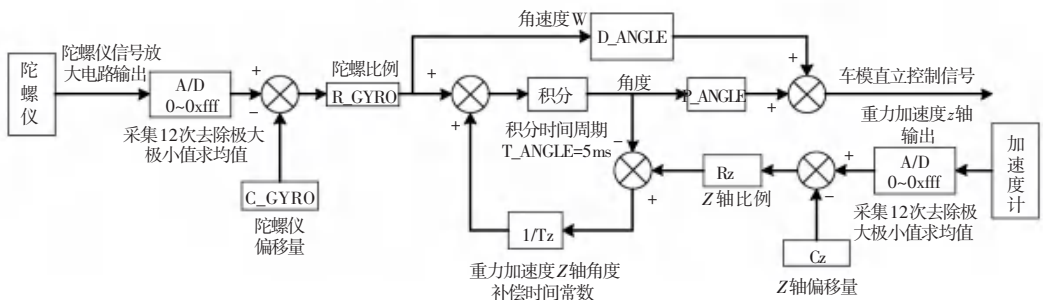


图 8 小车姿态算法设计框图

Fig. 8 Block diagram of the car attitude algorithm

### 2.1.4 通信协议设计

在下位机与上位机的通信过程中为了确保接收

的数据准确本控制系统还添加了简单的通信协议,其通信协议格式见表 1。

表 1 通信协议格式

Tab. 1 Communication protocol format

帧头为 AA	小车运行时偏离轨道的偏差值	小车运行时的姿态值	小车运行时的速度值	帧尾为 BB
占用 2 个字节	占用 3 个字节	占用 6 个字节	占用 5 个字节	占用 2 个字节

通信协议子程序的实现如下。

```

/* * * * * * * * * * * 将偏差、角度以及速度发送到上位机 * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * 帧头为 AA 占用 2 个字节,帧尾为 BB 占用 2 个字节 * * * * * * * * * *
#include "uart.h"

```

```

unsigned char Data[18];
void apart_num( int Data_Dy, float Data_Angle, int Data_Speed)
{
int i = 0;
Data[0] = 'A';

```

```

Data[1] = 'A';
if(Data_Dy>0)
{
Data[3] = '+';
Data[4] = Data_Dy/10+0x30;
Data[5] = Data_Dy%10+0x30;
}
else
{
Data[3] = '-';
Data[4] = Data_Dy/10+0x30;
Data[5] = Data_Dy%10+0x30;
}
if(Data_Angle>0)
{
// Data_Angle = Data_Angle * 1000;
Data[6] = '+';
Data[7] = (int)Data_Angle/10000+0x30;
Data[8] = (int)Data_Angle/1000%10+0x30;
Data[9] = (int)Data_Angle/100%10+0x30;
Data[10] = (int)Data_Angle/10%10+0x30;
Data[11] = (int)Data_Angle%10+0x30;
}
else
{
Data_Angle = Data_Angle * 1000;
Data[6] = '-';
Data[7] = (int)Data_Angle/10000+0x30;
Data[8] = (int)Data_Angle/1000%10+0x30;
Data[9] = (int)Data_Angle/100%10+0x30;
Data[10] = (int)Data_Angle/10%10+0x30;
Data[11] = (int)Data_Angle%10+0x30;
}
Data[12] = Data_Speed/1000+0x30;
Data[13] = Data_Speed/100%10+0x30;
Data[14] = Data_Speed/10%10+0x30;
Data[15] = Data_Speed%10+0x30;
Data[16] = 'B';
Data[17] = 'B';
for(i=0; i<=17; i++)
{
Uart_SendChar(UARTR0, Data[i]);
}
}

```

## 2.2 上位机软件设计

上位机软件实现了对 PC 机 IP 地址的获取、通信协议的处理以及对所获取的数据进行处理与显示。如图 9 所示为上位机程序流程图。

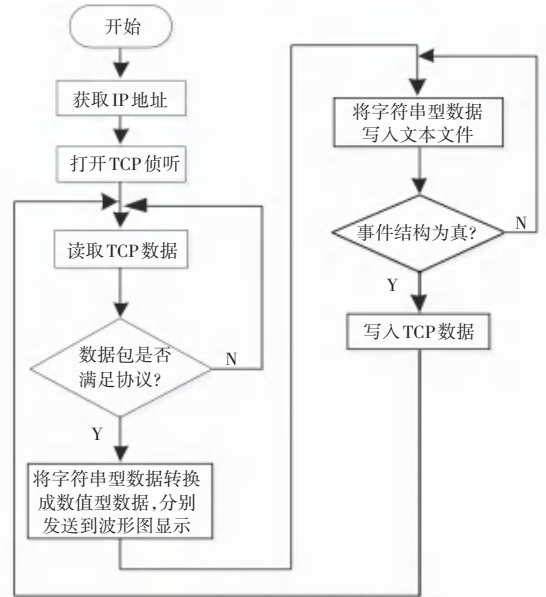


图9 上位机程序流程图

Fig. 9 Host computer program flow chart

(1) 上位机通过“字符串至 IP 地址转换”函数<sup>[2]</sup>获取当前机器的 IP 网络地址,然后在通过“IP 地址至字符串转换”函数转换成字符串,然后将其输入到字符串显示控件。当该上位机运行时即可获得当前 PC 机的 IP 地址<sup>[8]</sup>。

(2) 上位机通过读取 TCP 函数以及写入 TCP 函数实现了数据的接收与发送。

(3) 上位机通过“截取字符串”函数截取所获得数据包的前两个字节数据和后两个字节数据使用“等于?”函数判断其是否等于本设计所设定的帧头和帧尾,当其均相等时再通过“与”逻辑函数进一步判断是否为真,如若为真则通过事件结构处理接收的数据,通过“字符串截取”函数将每个字节所对应的数据截取到相应的图表显示出来。

## 3 结束语

本文设计了以 S9KEAZ128AMLK 芯片为核心处理器的下位机及以 LabVIEW 为上位机的智能小车控制系统,实现了智能小车的快速循迹以及上位机与下位机之间的局域网通信,并通过通信协议实现了智能小车传感器数据的实时监控以及智能小车的实时控制。经实验表明,该智能小车控制系统运行稳定,效果良好。(下转第 158 页)