

doi:10.16018/j.cnki.cn32-1650/n.201803007

基于 MSP430 单片机的远程直流电机控制系统设计

陈 中¹,杨汉华²

(1. 盐城工学院 电气工程学院,江苏 盐城 224003;2. 盐城工学院 学生处,江苏 盐城 224003)

摘要:基于 MSP430 单片机的 SPI 总线寄存器的通信控制方式,在对远程控制直流电机控制系统中硬件系统与软件系统的分析与设计的基础上,以无线模块 nFR24L01 为桥梁,充分利用无线模块 nRF24L01 可以一次发送并接收多帧信息的特点,进行控制系统和响应系统的 SPI 通信以实现远程控制直流电机正确运转的目的。最后通过实验证明了本文提出方案的正确性。

关键词:微处理器 远程控制 参数变化 SPI 通信

中图分类号:TM341 **文献标识码:**A **文章编号:**1671-5322(2018)03-0034-05

直流电机调速控制系统包括控制系统和响应系统。目前大多数直流电机的控制系统与响应系统采用导线直接连接,故障率较高。为了克服传统直流电机控制系统的这种缺点,本文基于 MSP430 单片机在直流电机控制系统和响应系统间采用无线信号传输进行远程控制。该系统不仅可以在恶劣环境或人工不方便的环境中工作,实现对直流电动机的远程控制,而且通过无线通信一次发送多帧信号,提高了传输效率,降低了控制系统的成本和故障率,具有较高的实用价值。

1 基于 MSP430 单片机远程直流电机控制的工作原理

基于 MSP430 单片机的远程直流电机控制系统由控制系统和响应系统两部分组成,包括无线通信模块、MSP430 单片机、直流电机驱动模块、矩阵式键盘等,如图 1 所示。其工作原理是:控制系统通过矩阵式键盘设定直流电机运行时间、高低速和正反转数值,并显示在液晶 12864 上,电机运行时,每过 1 s,液晶 12864 显示的数值减 1;按下发射键后,无线通信模块一次把多帧信息同时发送到响应系统,同时液晶 12864 实时显示当前时间;响应系统由 H 桥电路、无线模块、数码管和直流电机等组成,响应系统收到信号后,对多帧信号进行相应的处理,也就是根据控制系统设定的

直流电机运行时间、高低速和正反转情况通过驱动电路控制直流电机的运行,同时电机旋转时间显示在数码管上,且每过 1 s,数码管显示数值减 1,当数码管显示时间到 0 时电机停止运转(作用就是相当是定时器)^[1]。

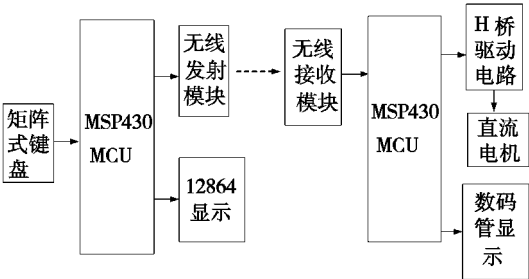


图 1 系统结构框图

Fig. 1 The structure diagram of system

需要说明的是,由于无线模块传输最大数据是 8 位二进制,所以对于高于 256 s 的设定时间,在控制系统和响应系统内部都必须进行相应的处理^[2]。

2 基于 MSP430 单片机的直流电机远程控制系统的电路设计

2.1 SPI 通信电路的设计

基于 MSP430 单片机直流电机远程控制系统包括控制系统和响应系统,其电路图分别如图 2、图 3 所示。由于系统的控制系统和响应系统都采

用 nRF24L01 芯片,该芯片采用全双工模式、SPI 总线方式^[1]。控制系统通过无线模块一次发送多帧数据,响应系统通过无线模块一次接受多帧数据。MSP 单片机具有 SPI 总线接口,其与无线

模块连接方式如下:nRF24L01 芯片的 MOSI、MISO 分别接单片机的 P3.1 口和 P3.2 口,IRQ、SCK、CSN 和 CE 分别接单片机的 P1.7、P3.3、P3.0和 P1.6 口。

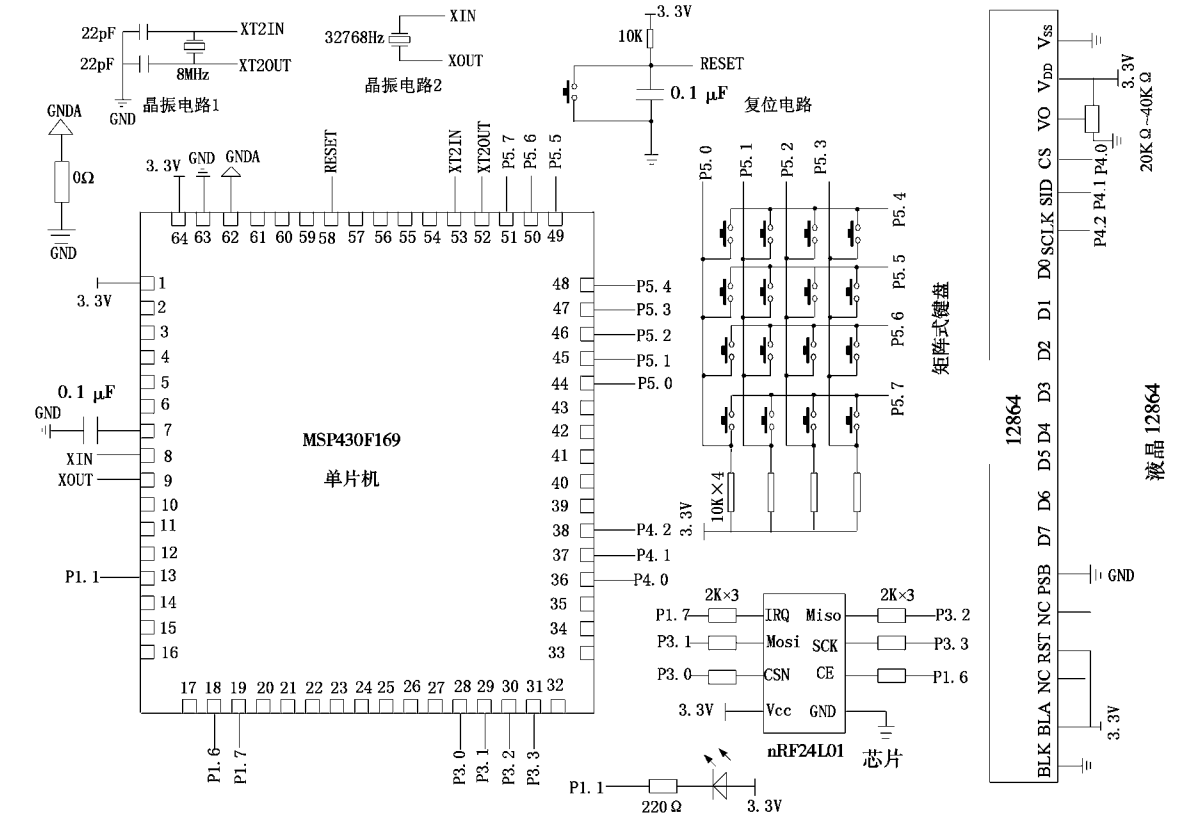


图2 控制系统电路图

Fig. 2 The circuit diagram of control system

2.2 控制系统的矩阵式键盘电路和显示电路的设计

基于 MSP430 单片机直流电机的控制系统中矩阵式键盘电路用于实现设定直流电机的运行时间、高低速和正反转等功能。该键盘采用 4 * 4 矩阵式键盘,其中低于 10 的为数字键,大于 10 的为功能键。当设定好运行时间、高低速和正反转后,通过功能键一次发送多帧信号^[3],这些信号经无线发射模块远程传输到响应系统中的无线模块,再经 MSP430 单片机处理后控制电机的运行。矩阵式键盘与单片机的连接方式如下:键盘的行和列分别接单片机的 P5.0 ~ P5.3 口和 P5.7 ~ P5.4 口。

2.3 响应系统的电机驱动电路和显示电路的设计

基于 MSP430 单片机响应系统的四位数码管显示直流电机运行时间,数码管段选经限流电阻

220 Ω 接单片机的 P6.7 ~ P6.0 口。电机驱动电路由 4 个开关管(并联续流二极管)组成,控制电机高低速和正反转。由于直流电机的高低速是通过改变 PWM 波的占空比实现的^[4],而 MSP430 单片机对于 PWM 波有着对应的端口。本设计采用定时器 B 产生 PWM 波,开关管对角分别接单片机的 P4.2 口和 P4.3 口。改变占空比就能改变电机转速。

液晶 12864 采用串行通信方式,其与单片机连续方式为:液晶 12864 的 SCLK、SID、CS 分别接单片机的 P4.2 ~ P4.0 的口。

3 软件系统设计

3.1 控制系统程序

控制系统软件主要由键盘扫描程序设定信息、发送程序、12864 液晶显示程序和采用寄存器


```
}
return keyvalue;
.....
主程序对键值的不同处理程序如下。
if( key_shu < 10)
{
    shu5 = shu5 * 10 + key_shu;
}
.....
if( key_shu == 11) ////正转
{
    state = 1;
    TX_BUF[0] = 1;
    SPI_RW_Reg ( WRITE_REG + STATUS,
0XFF);
    lcd_pos(2,0);
    hzkdis( tab4);
}
else if( key_shu == 12)////反转
{
    state = 1;
    TX_BUF[0] = 0;
    SPI_RW_Reg ( WRITE_REG + STATUS,
0XFF);
    lcd_pos(2,0);
    hzkdis( tab5);
}
if( key_shu == 13) ///高速
{
    TX_BUF[1] = 1;
    SPI_RW_Reg ( WRITE_REG + STATUS,
0XFF);
    lcd_pos(2,6);
    hzkdis( tab2);
}
else if( key_shu == 14) ///低速
{
    TX_BUF[1] = 2;
    SPI_RW_Reg ( WRITE_REG + STATUS,
0XFF);
    lcd_pos(2,6);
    hzkdis( tab3);
}
if( keyvalue == 10)
```

```
.....
if( key_shu < 10) {.....} 里面的语句是设定
直流电机运转时间(s);if( num1 == 11) {.....}
和 if( num1 == 12) {.....} 里面的语句是设定直
流电机旋转方向,并在液晶 12864 上显示;
if( num1 == 13) {.....} 和 else if( num1 == 14)
{.....} 里面的语句是设定直流电机高低速,并
在液晶 12864 上显示。以上这些信息将同时存入
无线模块寄存器中。
if( num1 == 10) {.....} 是发送键。当设定
好时间、方向和高低速后,按下发送键即可完成信
息的传输。
3.1.2 液晶显示和定时程序
由于电机运转时间需要实时显示,当经过了
1 s 后,时间减 1 并显示。但液晶 12864 显示时间
的位数发生改变时,其寄存器显示的位数保持不
变,从而出现显示错误[6]。比如,当液晶 12864 从
10 s 变化到 9 s 时,会显示 90。解决的方法是给
液晶 12864 寄存器显示的位数打上“补丁”。
寄存器个位显示语句如下:
if( num < = 9&&num > = 0)
{
    dis_flag = 1;
    table[0] = num% 10 + 0;
    table[1] = ``;
}
for( k = 0; k < 2; k + + )
{
    write_dat( table[ k] );
    delay(5);
}
.....
}
在寄存器显示的位数打上“补丁”后,其个位
显示具体的数值,但十位却赋值“空格”(注意要
加上英文单引号),这样当时间从 10 变为 9 时,寄
存器个位就显示 1 位数 9,末两位显示的不是 2
位数 90。其他位显示语句同理。
if( ( shu5 > 0) && ( time_1s_ok == 1) ) { time_
1s_ok = 0; shu5 - -; }
Shu5 = shu5 - - 是液晶 12864 上实时显示当
前时间的语句,if( time_1s_ok == 1) 是定时器产生
1 s 时间的语句。当定时到 1 s,过了 1 s 后液晶
```

12864 显示的数字为当前数字减 1。

定时 1 s 的定时器函数如下:

```
#pragma vector = TIMERB0_VECTOR
__interrupt void Timer_B( void)
{
    bb + + ;
    if( bb == 133 )
    {
        bb = 0;
        time_1s_ok = 1;
    }
}
```

采用定时器 B0 捕捉比较方式。在时钟函数 void Clk_Init() 上设置时钟频率为 8 MHz, 在主程序里设置计数器 TBCCR0 = 60 000, 计数器, 使其每增加 1, 即经过 $1/8 \mu\text{s}$, 在主程序里设置 TBCTL = CNTL_0 + TBSEL_2 + MC_1 语句说明计数的终值是 TBCCR0, 那么计数到终值时需要 $1/8 \mu\text{s} \times 60\,000$, 重复 133 次(if(bb == 133)), 就是 $1/8 \mu\text{s} \times 60\,000 \times 133$, 即近似 1 s。

3.1.3 SPI 通信程序

系统采用寄存器方法进行 SPI 总线通信。通信时首先进行端口初始化^[6], 包含使能 USART0 SPI 模式、8 位数据传输、3 线 SPI 模式和波特率设定等。

其部分程序如下:

```
void RF24L01_IO_set( void)
{
    ME1 | = USPIE0; // 使能 USART0 SPI 模式
    UCTL0 | = CHAR + SYNC + MM; // 8 位数据, SPI 模式, 主机模式
    UTCTL0 | = CKPH + SSEL1 + SSEL0 +
    STC; // SMCLK 时钟信号, 3 线 SPI 模式
    UBR00 = 0x02; // 波特率设定
    UBR10 = 0x00; // 波特率设定
    .....
}
```

3.2 响应系统程序

响应系统程序包括电机运行程序、SPI 通信程序以及数码管显示程序。电机运行是根据控制系统发来的信息(电机运行时间、高低速和正反转)进行的; SPI 通信程序是完成与控制系统之间的通信, 主要是一次接收从控制系统发过来的多帧信息(电机运行时间、高低速和正反转)^[7]; 数

码管主要是显示电机的运行时间, 每经过 1 s, 数码管上数字减 1, 到达设定的定时时间时电机停止运转。

在控制系统的发射程序里, 将正反转的设置放入 RX_BUF[0]、高低速的设置放入 RX_BUF[1]、运行时间的设置放入 RX_BUF[2], 当按下发射键(if(keyvalue == 10)) 后, 3 个信息(RX_BUF[0]、RX_BUF[1]、RX_BUF[2]) 依次发送。而在响应系统的接收程序, 这 3 个信息将依次放入 direction、speed、time 变量, 即 direction = (RX_BUF[0]);

speed = (RX_BUF[1]); time = (RX_BUF[2]), 然后进行相应的处理^[8]。

上述程序中, if((direction == 0) && (speed == 0)) { } 是直流电动机正转低速运行的程序。当运行这条语句时候, 打开定时器 B 中断(TBCCTL0 = CCIE), TBCCTL2 对应的单片机引脚是 P4.2, 定时器 B 产生的 PWM 波控制 H 桥对角三极管基集, 而 TBCCTL3 对应单片机引脚是 P4.3 定时器 B, 产生的 PWM 波控制 H 桥的另一对对角三极管的基集。两者占空比相同(TBCCR2 = 1000; TBCCR3 = 1000), 但 PWM 波方向相反(TBCCTL2 = OUTMOD_7; TBCCTL3 = OUTMOD_2);。

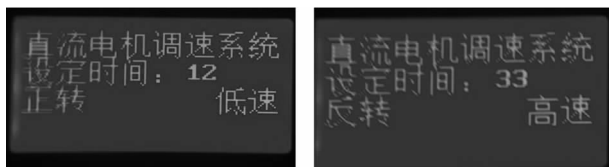
if((direction == 1) && (speed == 0)) { } 等语句同理, 只不过占空比改变, 将改变电机的平均电压, 从而改变电机转速。PWM 波方向改变时, 电机旋转方向随之改变^[9]。

else if(shu2 == 0) 语句是电动机停止语句。当液晶显示的数值为 0 时, 占空比均为 0, 即 TBCCTL2 = OUTMOD_0; TBCCTL3 = OUTMOD_0, 此时电动机停止。

4 实验验证

为了验证以上分析结果, 根据系统硬件和软件设计, 做成实物进行试验, 结果如图 5 所示。

图 5a 是控制系统矩阵式键盘按下数字“12”, 设定正转和低速后再按发送键, 控制板上无线模块发送多帧信息, 响应系统的无线模块一次接收多帧信息, 直流电机按照设定的信息进行运行的效果图。控制系统上的液晶和响应系统上的数码管同时显示“12”(图中只标明控制系统液晶 12864 的显示), 且每经过 1 s, 显示数字减 1。当显示数字为零时, 直流电机停止。



a 正转低速 b 反转高速

图 5 SPI 实验效果图

Fig. 5 Experimental effect diagram of SPI

重新设定新的信息(定时 33 s、反转和高速,如图5b),响应系统自动刷新并重新开始按照指

定的信息运行,从而验证系统设计的正确性。

5 结论

基于 MSP430 单片机,在对远程控制直流电机控制系统中硬件系统与软件系统的分析与设计的基础上,充分利用无线模块 nRF24L01 可以一次发送并接收多帧信息的特点,采用 MSP430 单片机的寄存器方式进行 SPI 通信,保证远程控制直流电机的正确运行。同时,通过实验验证了系统设计方案的正确性,从而为分析和设计基于 MSP430 单片机的远程控制提供了新的方法。

参考文献:

- [1] 隋绍勇,郑维广,张振邦. 基于 STM32F103 和 nRF24L01 的近程无线数传系统设计[J]. 电子元器件应用,2010(12): 15-17.
- [2] 陈中,沈翠凤. 基于单片机直流调速控制系统设计[J]. 盐城工学院学报(自然科学版),2014,29(5):120-126.
- [3] 贾至江,孟令军,高世明. 基于 STM32 的无线传感器网络声定位节点的设计与实现[J]. 传感器与微系统,2016,30(21):120-122.
- [4] 刘蕴红,杨君宝. 基于 myDAQ 的直流电机 PWM 远程控制系统设计[J]. 电子设计工程,2014,22(2):111-113.
- [5] 王平. 基于单片机 I/O 口模拟的 SPI 串行通信实现[J]. 电子世界,2013(14):24.
- [6] 胡燕清. 基于 PIC18F452 单片机 SPI 通信的仿真设计[J]. 电子设计信,2016,24(7):176-179.
- [7] 陈铭. 单片机 89C52 的 I/O 模拟串口与 PC 机通信数据采集实现方法[J]. 中国水运,2009,9(11):109-110.
- [8] 陈中,朱代中. 基于 STC89C51 单片机的控制系统设计[M]. 北京:清华大学出版社,2015.
- [9] 陈中,陈冲. 基于 MSP430 单片机的控制系统设计[M]. 北京:清华大学出版社,2017.

Design of Remote DC Motor Control System Based on MSP430 MCU

CHEN Zhong¹, YANG Hanhua²

(1. School of Electrical Engineering, Yancheng Institute of Technology, Yancheng Jiangsu 224003, China;
2. Students' Affairs Division, Yancheng Institute of Technology, Yancheng Jiangsu 224003, China)

Abstract: The communication and control mode of SPI bus register is based on MSP430 single chip microcomputer. On the basis of the analysis and design of hardware and software system in remote control DC motor control system, the wireless module nRF24L01 is used as a bridge, and the wireless module nRF24L01 can send and receive multi-frame information at one time. The SPI communication between the system and the response system is to achieve the goal of remote control of the DC motor running correctly. Finally, the correctness of the proposed scheme is proved by experiments. Finally, the correctness of the proposed scheme is proved by experiments.

Keywords: microcontroller; remote control; parameters changing; SPI communication

(责任编辑:李华云)