

实验三 电机驱动及转速测量实验

1 实验目的

- (1) 掌握单片机通用 I/O 口的使用；
- (2) 掌握使用单片机定时器产生占空比可调的 PWM 波；
- (3) 掌握使用单片机定时器 2 的捕获功能实现电机转速测量的方法。

2 实验电路

2.1 实验装置硬件说明

电机控制实验箱的原理框图如图 1 所示。

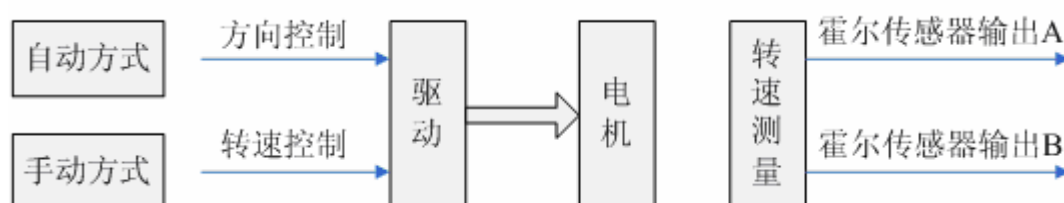


图 1 电机控制实验箱原理框图

从图 1 可知，控制电机需要两个信号，分别为方向控制和转速控制，控制信号经过驱动电路后驱动电机运转，在电机的轴上安装有塑料转盘，转盘上装有小磁铁，当电机运动时带动小磁铁运动，每个小磁铁经过安装的霍尔传感器后，霍尔传感器都会输出一个脉冲信号，这个脉冲信号在实验中用于检测电机的转速。

电机有两种控制方式，即手动方式和自动方式。在手动方式时，可以通过开关来控制电机方向；通过旋转电位器可以控制电机速度。在自动方式时，通过程序来控制电机的方向和转速。

2.2 实验装置接口说明

控制系统与电机实验箱通过 DB9 插头连接，其接口定义如表 1 所示。

表 1 电机实验箱 DB9 插头引脚信号特性

| DB9 引脚号 | 颜色 | 标号 | 信号特性 | 信号类型 | 信号方向 (对实验箱而言) |
|---------|----|-----|------|------|------------------|
| 1 | 棕 | DIR | 方向控制 | 数字量 | 输入 |
| 2 | 红 | PWM | 转速控制 | 数字量 | 输入 |
| 3 | 橙 | | | | |
| 4 | 黄 | | | | |

| | | | | | |
|---|---|------|-----------|-----|----|
| 5 | 绿 | GND | 电源地 | 地 | |
| 6 | 蓝 | OUTA | 霍尔器件 A 输出 | 数字量 | 输出 |
| 7 | 灰 | OUTB | 霍尔器件 B 输出 | 数字量 | 输出 |
| 8 | 白 | | | | |
| 9 | 黑 | | | | |

3 开发环境

程序开发调试软件为 KeilC，下载软件为 S51ISP，关于这两个软件的使用方法请参考“键盘显示实验指导书”。

4 实验要求

(1) 通过实验箱上的键盘输入调整 PWM 波的占空比，具体要求如下：

- 当按键为 0 时，其占空比为 20%，LED1 显示值为 0；
- 当按键为 1 时，其占空比为 40%，LED1 显示值为 1；
- 当按键为 2 时，其占空比为 60%，LED1 显示值为 2；
- 当按键为 3 时，其占空比为 80%，LED1 显示值为 3；

(2) 将测量到的电机转速显示到实验箱的数码管 LED3~LED6 上，转速单位为“转/分”。

5 例程参考

```
sbit PwmOut=P1^1;           //定义 PWM 的输出口为单片机的 P1.1
```

```
long data PwmH;              //PWM 高电平时个数
```

```
long data PwmL;              //PWM 低电平时个数
```

```
uchar PulsNum=0;             //脉冲数计数器
```

```
unsigned int OverFlow=0;      //定时器 2 溢出次数计数器
```

```
/******
```

```
电机初始化函数
```

```
*****/
```

```
void InitMotor(void)
```

```
{
```

```

    PwmOut=0;          //输出低电平

    PwmH=1500;          //PWM 高低电平各定时 1500 个数,即占空比为 50%

    PwmL=1500;

}

/*****

定时器初始化函数

*****/

void InitTimer()

{

    TMOD=0x01;          //定时器 0 工作于方式 1

    TH0=65535/256;      //定时器 0 计数初值设置

    TL0=65535%256;

    ET0=1;              //定时器 0 中断允许

    T2CON=0x09;          //定时器 2 工作于捕捉方式

    TH2=0x00;           //定时器 2 计数初值设置

    TL2=0x00;

    ET2=1;              //定时器 2 中断允许

    PT2=1;              //定时器 2 中断优先级最高

    TR0=1;              //启动定时器 0

    TR2=1;              //启动定时器 2

}

/*****

PWM 产生函数

```

```

*****/

void timer0() interrupt 1          //T0, 产生 PWM 波

{

    if(PwmOut==1)                //当前为高电平

    {

        TH0=(65536-PwmL)/256;    //计数值赋为低电平时间值

        TL0=(65536-PwmL)%256;

        PwmOut=0;                //输出低电平

    }

    else if(PwmOut==0)           //当前为低电平

    {

        TH0=(65536-PwmH)/256;    //计数值赋为高电平时间值

        TL0=(65536-PwmH)%256;

        PwmOut=1;                //输出高电平

    }

}

/*****/

定时器 2 捕获功能函数

*****/

void time2() interrupt 5

{

    if(EXF2==1)                  //捕获引起的中断

    {

        PulsNum++;              //脉冲个数加 1

        if(PulsNum==1)          //定时器清零

        {

            OverFlow=0;

            TH2=0;

            TL2=0;

        }

    }

}

```

```

    }

    else if(turn==9)          //电机转一圈后计算转速

    {

        TR2=0;

        datal=RCAP2L;          //读取捕捉值

        datah=RCAP2H;

        CaculateSpeed(datah,datal); //计算转速

        TR2=1;

        PulsNum=0;             //脉冲个数清零

    }

    EXF2=0;                   //清中断标志

}

else if(TF2)

{

    OverFlow++;               //溢出次数加一

    TF2=0;                    //清中断标志

}

}

```

```

/*****

```

转速计算函数

```

*****/

```

```

void CaculateSpeed(uchar a, uchar b)

```

```

{

    //速度=每分钟计数值/(溢出计数值+电机转一圈的计数值)

    speed=CountPerMinute/(65536*OverFlow+256*a+b);

}

```

6 实验报告

(1) 绘出硬件原理图

- (2) 给出软件流程图
- (3) 写出实验步骤
- (4) 附上带注释的软件源码，并对各模块进行说明
- (5) 总结实验心得