

2011 年全国大学生电子设计竞赛

基于自由摆的平板控制系统（B 题）

【本科组】



2011 年 9 月 2 日

摘 要

本题要求设计一个自由摆平板控制系统，其中倾角传感器 SCA60C、步进电机和单片机 AT89C52 系统是核心部件。利用 SCA60C 倾角传感器进行数据采集；在控制平板转动角度、速度方面，经过充分比较、论证，最终选用了控制精确的步近电机 42BYG250-40，其最小步进角 1.8 度，易于控制平板转动角度；将倾角传感器采集数据进行 AD 转换将数字信号传给单片机并通过共阳数码管实现显示自由摆偏转角度；采用单片机内部时钟实现精确计时。最后的实验表明，系统完全达到了设计要求，并增加了角度显示创新功能。

关键词：自由摆、倾角传感器 SCA60C、单片机、步进电机、数码管显示。

目录

一：系统方案	P4
1.1 方案论证	P4
1.1.1 控制器模块	P4
1.1.2 角度检测模块	P4
1.1.3 电机模块	P4
1.1.4 驱动电机模块	P4
1.1.5 显示模块	P4
1.1.6 电源模块	P4
1.2 系统设计	P5
1.3 结构框图	P5
二、理论分析与计算	P5
2.2 自由摆偏离角与平板倾角关系	P5
2.1 时钟周期与平板转动角度关系	P5
三、电路与程序设计	P6
3.1 检测电路设计	P6
3.2 电源模块设计	P6
3.3 驱动步进电机模块	P6
3.4 软件设计与工作流程图	P7
3.4.1 软件设计	P7
3.4.2 工作流程图	P7
四、测试方案与测试结果	P8
4.1 测试仪表	P8
4.2 自由摆摆动一个周期时间测试	P8
4.3 平板转动固定角度需给脉冲数目的确定	P8
4.4 测试功能实现	P8
五、结语	P8
附录	
附录一 总原理图	P10
附录二 源程序	P11

一：系统方案

本题要求设计并制作一个自由摆上的平板控制系统，能实现在自由摆摆动过程中平板自由旋转，以及在自由摆摆动过程中平摆上放置硬币使硬币不滑落。发挥部分要求实现用手推动摆杆至一个角度 θ (θ 在 $30^\circ \sim 60^\circ$ 间)，启动后，系统应在 15 秒钟内控制平板尽量使平板上的激光笔照射在设定的中心线上。

本系统利用角度传感器检测单摆摆动角度并用数码管显示，利用单片机接收信号并控制步进电机来控制平板转动。主要由角度检测模块、电机模块、驱动电机模块，显示模块，电源模块，控制器模块组成，下面分别论证这几个模块的选择。

1.1 方案论证

1.1.1 控制器模块

方案一：采用 ATMEEL 公司的 STC89C52。52 单片机价格便宜，应用广泛，其功能足以实现本设计要求。

方案二：采用凌阳公司的 SPCE061A 单片机作为控制器的方案。该单片机 I/O 资源丰富，但是价格昂贵。

综合考虑，我们选择价格便宜的方案一。

1.1.2 角度检测模块

方案一：采用深圳市华夏磁电子技术开发有限公司的 AME-B001 角度传感器，0—360 度测量范围，安装非常不方便，而且电压输出信号，采集不便。

方案二：采用 SCA60C 倾角传感器， $-90^\circ \sim +90^\circ$ 测量范围。模拟 0.5—4.5 输出，工作温度范围宽，价格便宜，安装方便，电压输出信号，采集不便。

方案三：采用 OMRON 公司的 E6B2-CWZ6C 编码器，数字信号输出，采集方便，价格昂贵，安装不便。

由性价比考虑以及安装简便程度选择方案二。

1.1.3 电机模块

方案一：采用直流减速电机控制平板，直流减速电机力矩大，转动速度快，但其制动能力差，无法达到平板转动的要求。

方案二：采用型号 42BYG 的步进电机控制平板的运动，最小步进角为 1.8 度，因此能实现平板转动的精确控制，而且当不给步进电机发送脉冲的时候，能实现自锁，从而能较好的实现平板停止转动的目的。

1.1.4 驱动电机模块

方案一：采用 BY-2HB03MA 步进电机驱动器，此款驱动器使用 THB6128 芯片，电源损耗低，开关效率高。细分数可选 1/1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128。

方案二：采用 ZD—6209—V2 步进电机驱动器，高可靠性，街口采用超高速光耦隔离，细分数可选 1/1, 1/2, 1/4, 1/8, 1/16。

考虑到本设计不需用太高的细分，1/8 细分即可，故选择方案二。

1.1.5 显示模块

方案一：用 LED 显示，优点亮度高、成本低。但不能显示汉字，显示内容较少。

方案二：采用金鹏电子的图形点阵式液晶 OCJM4*8C。串行接口，显示简单。

考虑到本设计显示模块只用来显示自由摆摆动的角度，数码管显示即可，故选方案一。

1.1.6 电源模块

方案一：用开关电源 S-25-12 高效率，高可靠，输出 12V 电压。

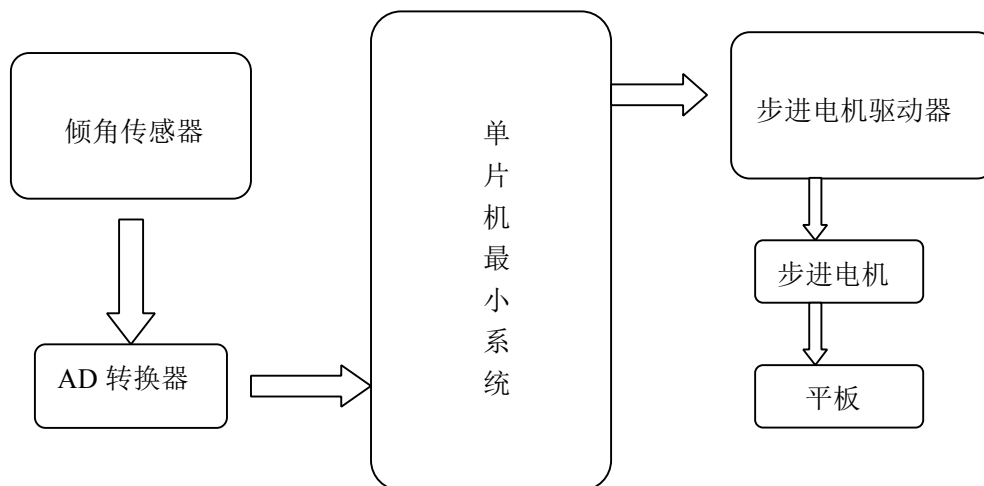
方案二：用电池组供电，相对较不可靠。

经比较,选择可靠性较高的方案一,输出的 12V 电源可直接给电压跟随器和驱动电机供电,其余部分所需的 5V 电源可利用三端稳压器稳压后输出。

1.2 系统设计

根据上述方案论证,我们确定以 STC89C52 作为控制核心,采用型号为 42BYG 的步进电机控制平板转动,采用 SCA60C 倾角传感器检测自由摆摆动角度,通过 AD 转换器将数字型号送给单片机,并利用数码管显示角度,采用 ZD—6209—V2 步进电机驱动器驱动步进电机。

1.3 结构框图



图一 系统总体结构框图

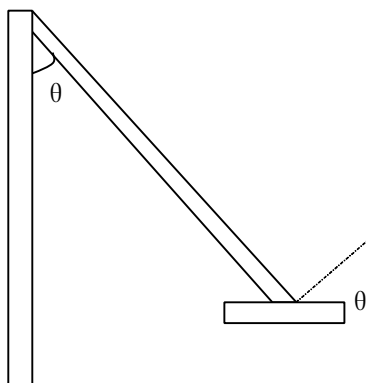
二、理论分析与计算

2.1 时钟周期与平板转动角度关系

本设计步进电机驱动器设定为 16 细分,1 细分时,步进电机驱动器步进角度为 1.8 度,故 16 细分时,每给一个脉冲步进电机转轴转动 $(1.8^\circ \div 16) = 0.1125^\circ$ 。

自由摆摆长为 100cm,由周期计算公式 $T = 2\pi \sqrt{L/g}$ 得出摆动周期大约为 2 秒。要使平板在自由摆摆动一个周期内旋转一周 (360°) 则单片机时钟周期为 312.5us

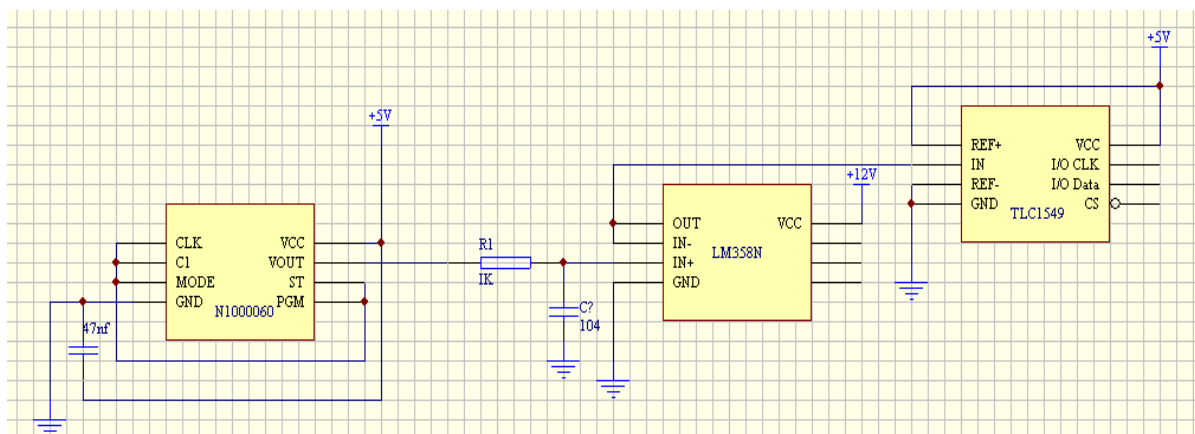
2.2 自由摆偏离角与平板倾角关系



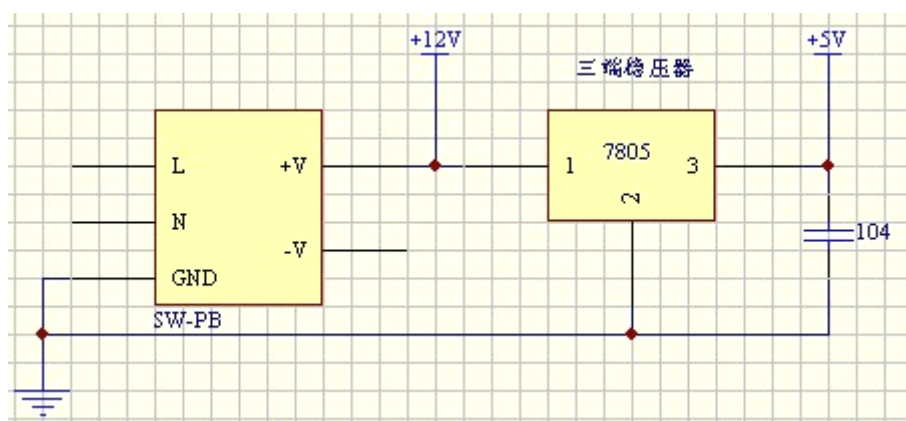
三、电路与程序设计

3.1 检测电路设计

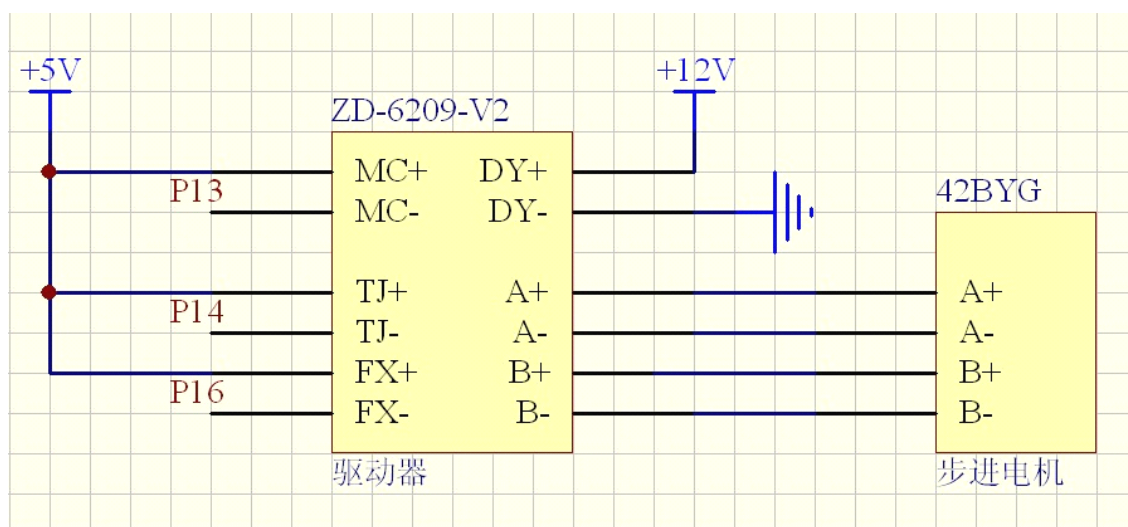
倾角传感器与电压跟随器连接,将电压信号传给 AD 转换器电路图如下所示。



3.2 电源模块设计



3.3 驱动步进电机模块

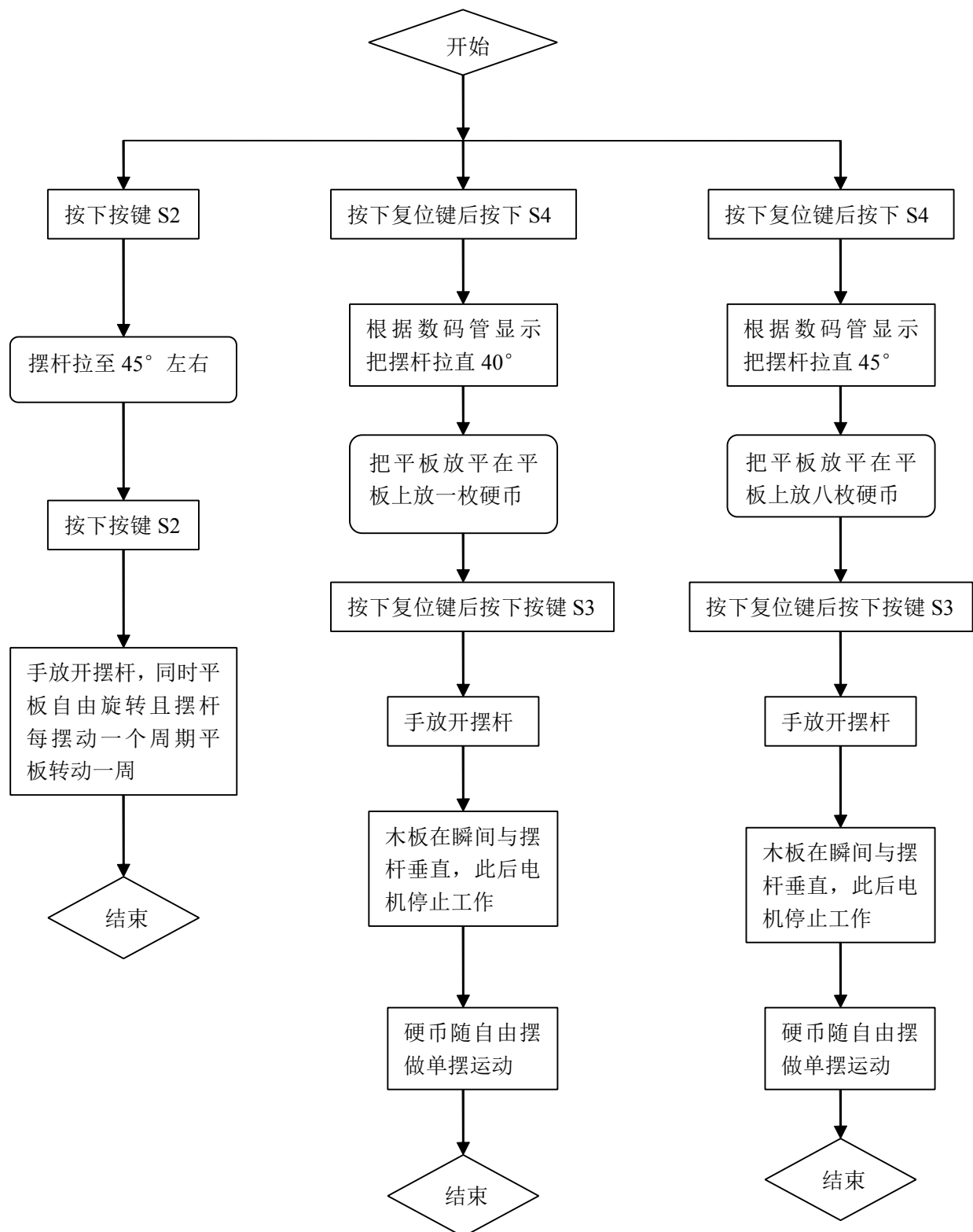


3.4 软件设计与工作流程图

3.4.1 软件设计

软件要实现功能如下：1、读倾角传感器角度；2、给步进电机步进脉冲；3、显示摆杆角度；4、键盘控制选择工作模式

3.4.2 工作流程图



四、测试方案与测试结果

4.1 测试仪表

4 位半数字万用表(MASTECH MY-65), 秒表, 计算器, 米尺。

4.2 自由摆摆动一个周期时间测试

根据理论公式自由摆摆动周期 $T=2\pi\sqrt{L/g}=2.0070s$ 实际试验中, 自由摆在摆动过程中顶端的轴承存在摩擦, 自由摆摆不到原最高位, 摆动周期会小。

表 1 自由摆摆动周期记录

次数	1	2	3	4	5
十个周期时间	19.76s	19.82s	19.74s	19.58s	19.69s

根据测试, 自由摆摆动一个周期时间大约为 1.9718s

4.3 平板转动固定角度需给脉冲数目的确定

要使平板转动一周, 根据理论可计算出所需脉冲数, 但是实际试验中, 步进电机转动时在与平板连接出有打滑现象, 经过反复试验, 数据记录如下, 其中, 设置驱动电机细分为 8 细分, 则每给一个脉冲步进电机转轴转动 0.225 度。

表 2 使平板转动一周所需脉冲数记录

脉冲数目	与平板转动 360 度差值
1600	-3.5
1610	-1.1
1613	-0.4
1615	+0.2

根据上述测试数据, 使平板转动一周所需脉冲数为 1615 个。

4.4 测试功能实现

表 3 基本功能 1 测试记录

次数	1	2	3	4	5
角度偏差	2.5°	3.2°	2.8°	3.4°	3.2

表 4 基本功能 2 测试记录

次数	1	2	3	4	5
偏离中心线	+0.3cm	+0.1cm	0	+0.3cm	0

表 5 基本功能 3 测试记录

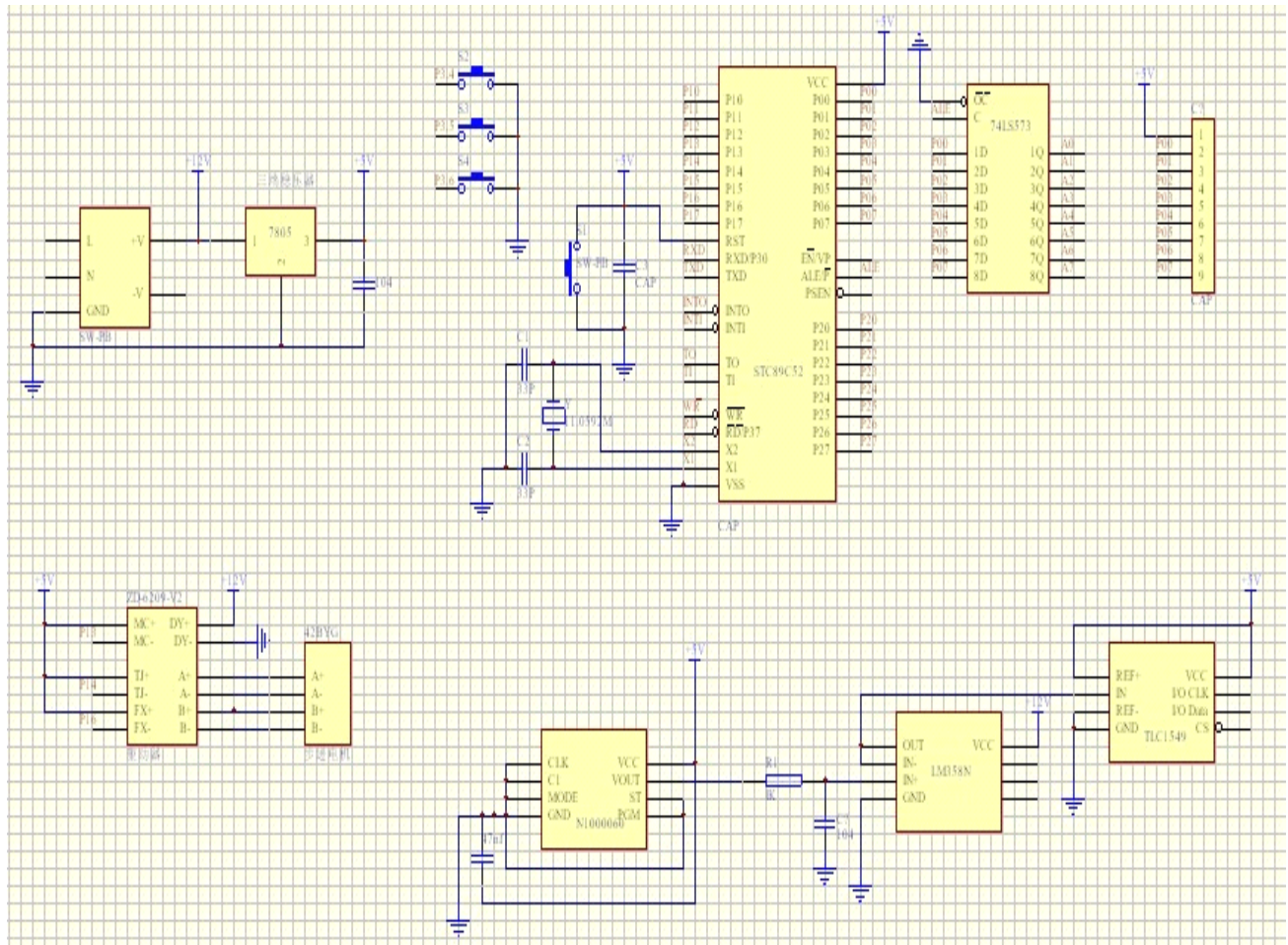
次数	1	2	3	4	5
掉落硬币数	0	0	2	0	0

五、结语

在这段时间内的试验过程中, 我们遇到过很多问题, 需要靠我们自己去考虑, 去动手, 去解决。平时我们多数时间只是在学习理论知识, 在准备比赛和进行比赛这段期间内, 我们将平时在课本上学习的 protel、keil、烧录软件等软件真真正正的运用到实践中。在设计过程中, 我们分工工作, 相互探讨, 相互学习, 相互监督, 增强了和他人的协作能力。我们更加增进了对团队工作意识, 增进了友谊。

然而，我们的电子设计作品还有很多不足，有待进一步的提高，在此恳请各位老师的批评指导。

附录一 总原理图



附录二 源程序

```
#include<reg52.h>
#include<intrins.h>
#include"math.h"
#define uchar unsigned char
#define uint unsigned int

sbit key1=P3^4;
sbit key2=P3^5;
sbit key3=P3^6;
sbit dula=P2^6;
sbit wela=P2^7;
sbit zx=P1^3;           //控制步进电机转向
sbit gz=P1^4;           //控制步进电机是否工作
sbit mc=P1^6;
sbit AD_CS = P2^4;       //设置单片机连接 TLC1549 的引脚
sbit AD_CLK = P2^2;
sbit AD_DAT = P2^1;      //控制步进电机脉冲

unsigned int adc_value1,adc_value2,adc_value3,adc_value4;
uchar tt,n;
uint duqu1,duqu2,duqu3,duqu4,cha,pinjun1,pinjun2,fre,num;
unsigned int adc_value;
uchar          code          table[]=
{0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90 };
uchar a,aa,bb,cc,dd,ee;
uint jiaodu;
float y,yy;

void display(uchar wan,uchar qian,uchar bai,uchar shi,uchar ge);
unsigned int readadc(void)           //读取 ADC 数据的函数
{unsigned char i;
unsigned int AD_Te,AD2,AD8;
AD8=0;
AD2=0;
AD_CS = 1;
AD_CLK = 0;
AD_DAT = 1;
AD_CS = 0;           //拉低 CS 信号，准备接收数据
for(i=0;i<2;i++)     //读取高两位
{AD_CLK =0;
_nop_();
AD2 =AD2<<1;
```

```

if (AD_DAT)
{AD2+=0x01;}
AD_CLK=1;
_nop_();
}
for(i=0;i<8;i++)          //读取低 8 位
{AD_CLK =0;
_nop_();
AD8 =AD8<<1;
if(AD_DAT)
{AD8+=0x01; }
AD_CLK =1;
_nop_();
}
AD_CLK =0;
AD_CS  =1;                //延时 21us 以上，等待转换完毕
AD_Te= AD2*256+AD8;      //合成 10 位整数
return (AD_Te);
}

void delay(uchar z)
{
    uint x,y;
    for(x=z;x>0;x--)
        for(y=110;y>0;y--);
}

void display(uchar wan,uchar qian,uchar bai,uchar shi,uchar ge);
void main()
{
    TMOD=0x01;
    TH0=(65536-250)/256;
    TL0=(65536-250)%256;

    if (key1==0)
    {
        mc=0;
        zx=1;

        EA=1;
        ET0=1;
        TR0=1;
        while(1)

```

```

        {
            readadc();
            adc_value1=readadc();
            readadc();
            adc_value2=readadc();
            readadc();
            adc_value3=readadc();
            readadc();
            adc_value4=readadc();

pinjun1=(adc_value1+adc_value2+adc_value3+adc_value4)/4;
            delay(4);
            readadc();
            duqu1=readadc();
            readadc();
            duqu2=readadc();
            readadc();
            duqu3=readadc();
            readadc();
            duqu4=readadc();
            pinjun2=(duqu1+duqu2+duqu3+duqu4)/4;
            cha=pinjun1-pinjun2;
            n=abs(cha);
            if(n>20)
                gz=1;
            else
                gz=0;

        }
    }
    if (key2==0)
    {

        num=0;
        gz=1;
        zx=0;
        mc=0;
        TMOD=0x01;
        TH0=(65536-250)/256;
        TL0=(65536-250)%256;
        EA=1;
        ET0=1;
        TR0=1;
    }

```

```

while(1)
{
    if (num==400)
    {
        num=0;
        TR0=0;
    }
    readadc();
    adc_value=readadc();
    if (100<adc_value&adc_value<495)
    {
        zx=1;
    }
    if (495<adc_value&adc_value<900)
    {
        zx=1;
    }
}

}

if(key3==0)
{
    unsigned int adc_value;
while(1)
{
    readadc();
    adc_value=readadc();
    y=0.225*(adc_value)-112.5;
    yy=abs(y);
    yy=yy*3/4;
    jiaodu=yy*1000;

    aa=jiaodu/10000;
    bb=jiaodu%10000/1000;
    cc=jiaodu%1000/100;
    dd=jiaodu%100/10;

    ee=jiaodu%10;

```

```

        for(a=10;a>0;a--)
        {
            display(aa,bb,cc,dd,ee);
        }
    }
}

```

```

}
void timer0() interrupt 1
{
    TH0=(65536-250)/256;
    TL0=(65536-250)%256;
    tt++;

    if(tt==2)
    {
        tt=0;
        mc=~mc;
        num++;
    }
}

void display(uchar wan,uchar qian,uchar bai,uchar shi,uchar ge)
{

```

```

    dula=1;
    P0=table[wan];
    dula=0;
    P0=0xff;
    wela=1;
    P0=0x02;
    wela=0;
    delay(1);

```

```

    dula=1;
    P0=table[qian];
    dula=0;
    P0=0xff;
    wela=1;
    P0=0x04;

```

```

    wela=0;
    delay(1);

    dula=1;
    P0=table[bai];
    dula=0;
    P0=0xff;
    wela=1;
    P0=0x08;
    wela=0;
    delay(1);

    dula=1;
    P0=table[shi];
    dula=0;
    P0=0xff;
    wela=1;
    P0=0x10;
    wela=0;
    delay(1);

    dula=1;
    P0=table[ge];
    dula=0;
    P0=0xff;
    wela=1;
    P0=0x20;
    wela=0;
    delay(1);

    dula=1;
    P0=0x7f;
    dula=0;
    P0=0xff;
    wela=1;
    P0=0x04;
    wela=0;
    delay(1);
}

```