

2015 年全国大学生电子设计竞赛

风力摆控制系统（B 题）



2015 年 8 月 15 日

摘 要

风力摆是一个复杂的快速、非线性、多变量、强耦合、自然不稳定的非最小相位系统,是对单摆圆锥摆问题的复杂化。该简易装置,运用物理学中基本单摆公式以及运动的合成分解,以及自动控制原理部分知识的建立了非线性离散化的数学模型。

本系统为由飞思卡尔 K60 单片机控制模块、姿态采集模块、风力摆模块、任务完成提示模块以及风力摆机械结构组成的闭环控制系统。MPU6050 采集风力摆姿态角,单片机处理姿态角数据后通过位置示 PID 闭环调节风力。通过实测数据及摆的关系依据香浓定理给出合适的采样离散时间,使其不发散。

功能上实现了风力摆在仅受直流风机为动力控制下快速起摆、画线、恢复静止的功能,并能准确画圆,且受风力影响后能够快速恢复画圆状态,并附加画出心形线,具有很好的鲁棒性,各参数及测试模式可由按键输入并通过响铃,智能性好,反应速度快。

关键词: PID 控制 香浓采样 K60 MPU6050 姿态解析

目 录

一、系统方案.....	1
1、动力系统的论证与选择.....	1
2、姿态测试仪的论证与选择.....	1
3、电机驱动的论证与选择.....	2
4、控制算法的论证与选择.....	2
二、系统理论分析与计算.....	2
1、直线上摆动的分析.....	2
2、任意角度摆动直线的计算.....	2
3、复杂摆动的计算.....	4
(1) 李萨如图形.....	4
(2) 圆形测画.....	4
(3) 附加心形线.....	4
三、电路与程序设计.....	5
1、电路的设计.....	5
(1) 系统总体框图.....	5
(2) 电机驱动子系统框图与电路原理图.....	5
(3) MPU6050 子系统框图与电路原理图.....	6
(4) 电源.....	6
2、程序的设计.....	7
(1) 程序功能描述与设计思路.....	7
(2) 程序流程图.....	8
四、测试方案与测试结果.....	9
(1) 测试结果(数据).....	9
(2) 测试分析与结论.....	9
五、结论与心得.....	10
六、参考文献.....	10
附录 1: 电路原理图.....	11
附录 2: 源程序.....	12

风力摆控制系统（B 题）

【本科组】

一、系统方案

本风力摆控制系统主要由单片机控制模块、电源模块、姿态采集模块、风力摆模块、人机交互显示系统以及风力摆机械结构组成。风力摆由万向节连接碳杆再连接风机组成。位于碳杆中部的姿态采集模块不断采集风力摆当前姿态角，并返回单片机。单片机处理数据后通过控制 PWM 波占空比控制风机转速，实现对风力摆的控制。下面分别论证这几个模块的选择。

1、动力系统的论证与选择

方案一：

采用 4 只直流电脑散热风机作为动力系统。四只风机取一边靠于摆杆，朝向成顺时针排列，通过控制四只风机转速控制风力摆当前状态。此方案风力摆负载最重，但自身过沉不适于控制大的摆动。

方案二：

采用 4 只四旋翼空心杯风机作为动力系统。四只风机十字排列，粘于自行设计的承载板并连接到杆上，通过控制四只风机转速控制风力摆当前状态。此方案风机转速大，重量轻，控制风力摆状态最为精确，且动力最足。

综合上述比较，考虑系统的快速工作以及精确控制，本系统采用方案二。

2、姿态测试仪的论证与选择

方案一：

只测量风力摆关于静止状态时的偏转角。采用二维平面内角位移传感器测量风力摆转动时关于静止状态时的偏转角，通过控制该偏转角实现对流风机的控制。该方案软件处理繁琐，且二维平面内的角位移传感器不利于测量风力摆的空间位置，不利于实现对风力摆的精确控制。

方案二：

选用双轴倾角传感器模块 LE-60-OEMLE-60-OEM，测量重力加速度变化，转为倾角变化，可测量双向。具有稳定性高、低功耗、结构简单等优点。响应速度为 5Hz。它可以测量平衡板与水平方向的夹角，x,y 方向可以测，但 z 轴不可测。且操作复杂，软件处理难度大。

方案三：

采用三维角度传感器 MPU6050。用三维角度传感器时刻测量风力摆当前姿态，通过处理采集的姿态角数据控制风机带动风力摆运动。此方案可精确测量风力摆当前姿态，实现对风力摆的精确控制。

综合以上三种方案，选择方案三。

3、电机驱动的论证与选择

方案一：

BTN7971 电机驱动。该方案经测试，发现带载能力一般，但对于本系统已足够，且便于搭建，电路简易。

方案二：

采用 IR2104 芯片驱动。该方案带载能力较强，功耗小，控制简单，但电路较复杂。综合以上二种方案，选择方案一。

4、控制算法的论证与选择

方案一：

采用模糊控制算法。模糊控制有许多良好的特性,它不需要事先知道对象的数学模型,具有系统响应快、超调小、过渡过程时间短等优点，但编程复杂，数据处理量大。

方案二：

采用 PID 算法。按比例、积分、微分的函数关系进行运算，将其运算结果用以输出控制。优点是控制精度高，对于本系统的控制已足够精确，节约了单片机的资源和运算时间。

综合比较以上两个方案，本系统选择方案二。

二、系统理论分析与计算

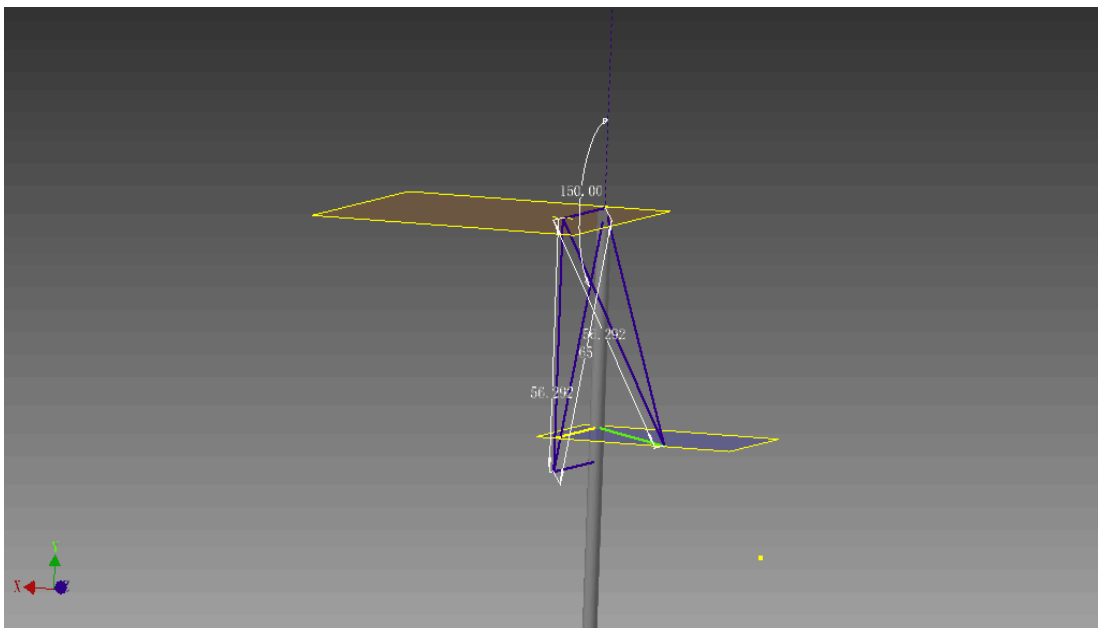
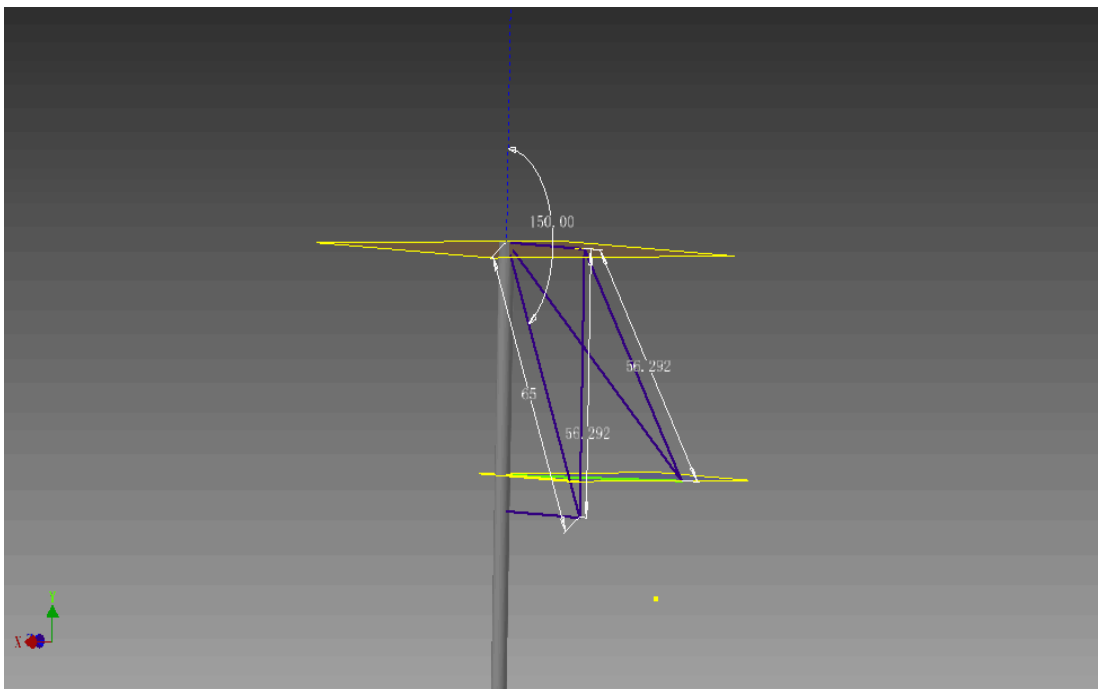
1、直线上摆动的分析

对于本系统的风力摆，假定一维下是近单摆运动，由于单摆的往复性，可根据正弦函数离散化后来设定不同时刻的目标值，用 PID 来进行跟随，由于有个较稳定的跟随状态以及离散化较稳定，不会出现发散的状况，对于不同的长度，正弦公式 $Y=A\sin\omega t$ 中的赋值 A 改变即可。

其中离散化后可用 t/T 周期代表式中转过的角度来给予相对应的长度完成单摆。

2、任意角度摆动直线的计算

对于任意角度，我们对系统建立三维模型如下：



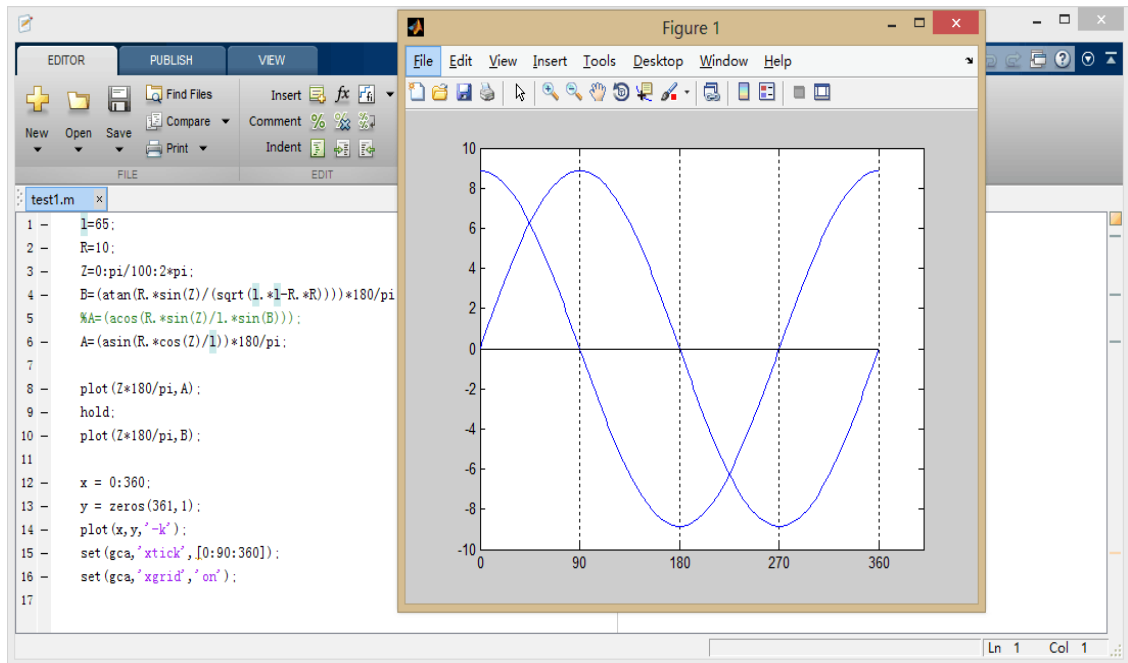
$$\text{SetAngle_X} = (\text{asin}(R \cdot \cos(Z/180 \cdot 3.14) / l)) \cdot 180 / 3.14;$$

$$\text{SetAngle_Y} = (\text{atan}(R \cdot \sin(Z/180 \cdot 3.14) / (\text{sqrt}(l^2 - R^2)))) \cdot 180 / 3.14;$$

其中：

R 为半径，L 为杆长度，经计算即可得到每次的不同角度直线的设定值。

在 MATLAB 下仿真：



与预估值一致。

3、复杂摆动的计算

(1) 李萨如图形

李萨茹曲线由以下参数方程定义：

$$x(\theta) = a \sin \theta$$
$$y(\theta) = b \sin(n\theta + \varphi)$$

若 $a=b$, $n=1$, 则曲线是椭圆。

若角度相差 90° , 则这椭圆其实是圆。

(2) 圆形测画

在实际程序中，我们将 x 方向给位 \sin 函数， y 方向给位 \cos 函数，由于系统本身的误差，所以对于同是开始的 x , y 运动不能给相同的赋值，因此实测后给与一个修正系数来使图像变得更加平滑，更加趋于目标图形。

(3) 附加心形线

对于附加功能，查阅到心形线的参数方程如下：

$$x=a*(2*\cos(t)-\cos(2*t))$$

$$y=a*(2*\sin(t)-\sin(2*t))$$

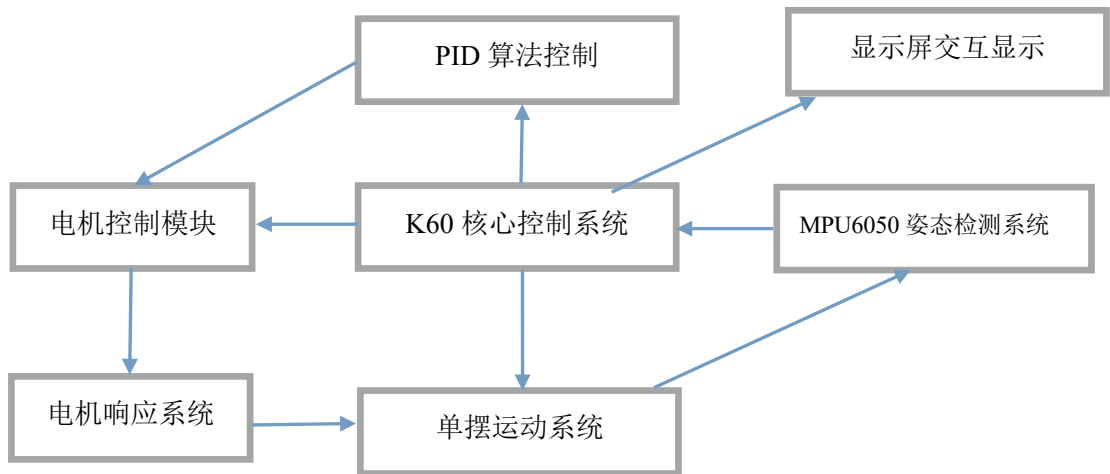
对于此心形线，采取更短的离散化时间，使图像不失真后，控制 PID 经过实测后完成效果。

三、电路与程序设计

1、电路的设计

(1) 系统总体框图

系统总体框图如图一所示



图一 系统总体框图

(2) 电机驱动子系统框图与电路原理图

电机驱动子系统框图

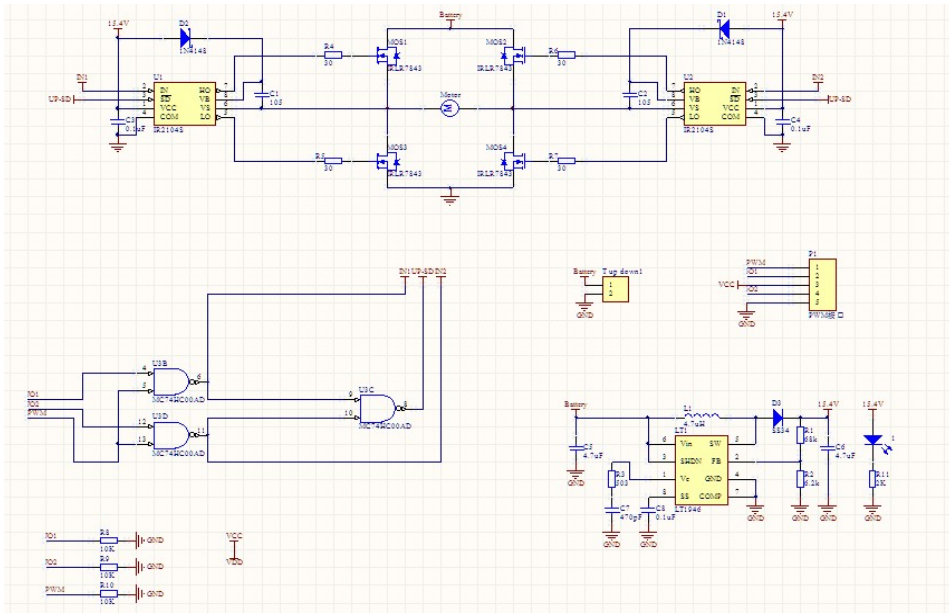


图 1 电机驱动子系统电路

(3) MPU6050 子系统框图与电路原理图

MPU6050 子系统电路

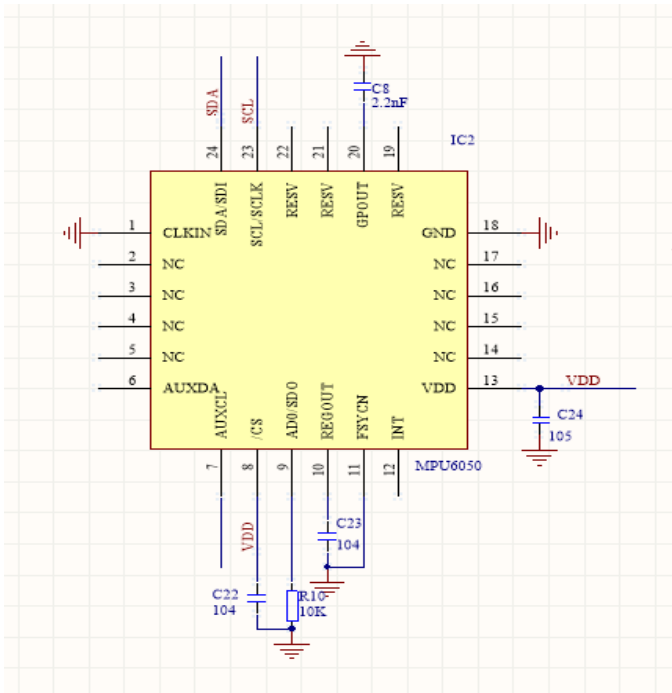


图 2 MPU6050 子系统电路

(4) 电源

电源由变压部分、滤波部分、稳压部分组成。为整个系统提供 5V 和 3.3V 电压，确保电路的正常稳定工作。这部分电路比较简单，采用 LT1529, LT1085 芯片。

2、程序的设计

(1) 程序功能描述与设计思路

1、程序功能描述

(1) 从静止开始，15s 内控制风力摆做类似自由摆运动，使激光笔稳定地在地面画出一条长度不短于 50cm 的直线段，其线性度偏差不大于 $\pm 2.5\text{cm}$ ，并且具有较好的重复性；

(2) 从静止开始，15s 内完成幅度可控的摆动，画出长度在 30~60cm 间可设置，长度偏差不大于 $\pm 2.5\text{cm}$ 的直线段，并且具有较好的重复性；

(3) 可设定摆动方向，风力摆从静止开始，15s 内按照设置的方向（角度）摆动，画出不短于 20cm 的直线段；

(4) 将风力摆拉起一定角度（ $30^\circ \sim 45^\circ$ ）放开，5s 内使风力摆制动达到静止状态。

(5) 以风力摆静止时激光笔的光点为圆心，驱动风力摆用激光笔在地面画圆，30s 内需重复 3 次；圆半径可在 15~35cm 范围内设置，激光笔画出的轨迹应落在指定半径 $\pm 2.5\text{cm}$ 的圆环内；

(6) 在圆周运动，在距离风力摆 1~2m 距离内用一台 50~60W 台扇在水平方向吹向风力摆，台扇吹 5s 后停止，风力摆能够在 5s 内恢复圆周运动，激光笔画出符合要求的轨迹；

(7) 画出心形线。

根据题目要求软件部分主要实现键盘的设置和显示。

1) 矩阵键盘实现功能：设置划线角度，长度，圆的半径。

2) 按键功能：归到原点，设置成功提示。

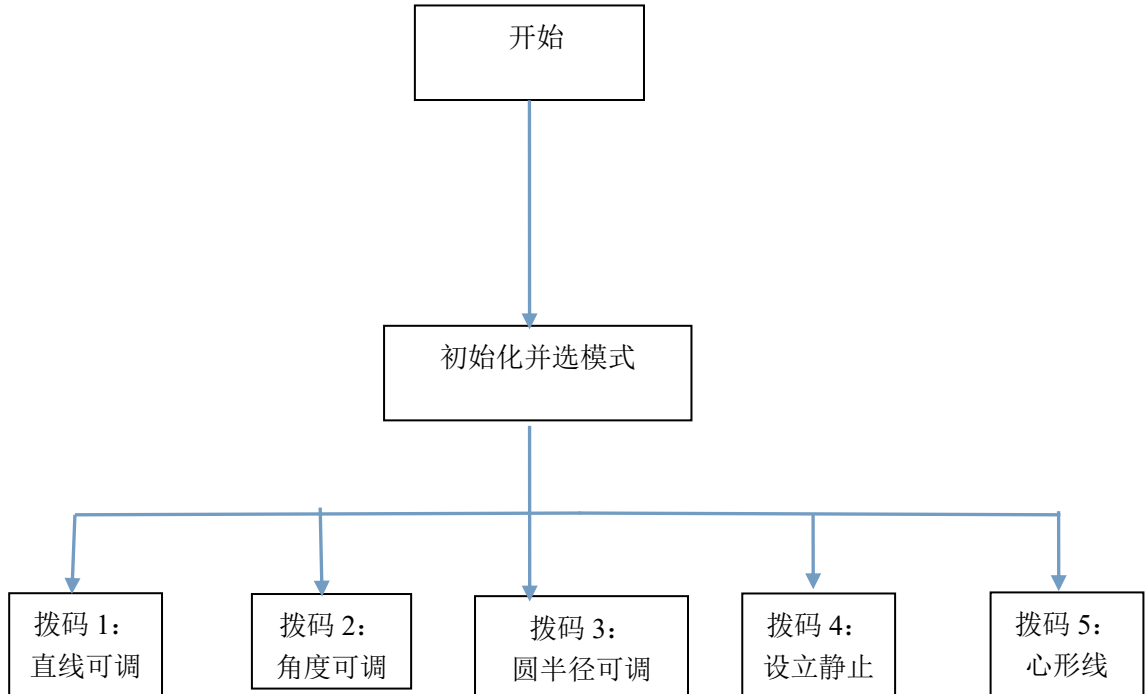
3) 拨码功能：设置不同功能的模式。

2、程序设计思路

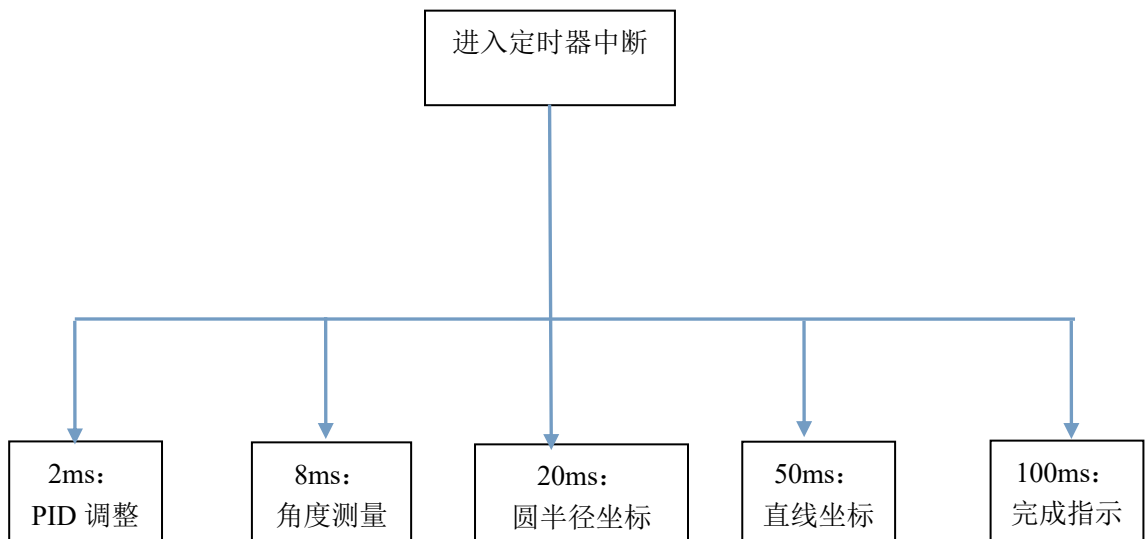
通过 mpu6050 测出的角度与设定值形成闭环来控制 PWM 控制四路电机的转速，同时为简便起见，将横竖设为两大组，转向调为一致。利用两大套的 PID 不同参数控制两个垂直方向的电机。对于设定的长度，经实测后拟合为函数写入程序，便于调节任意长度。对于不同的图形要求，取单摆的时长为自身周期，图形离散化周期依据香农采样定理为 2 倍图形周期，再经实测后微调，同时对每个电机的死区以及两个大组电机幅值微调，以达到最佳效果。

(2) 程序流程图

1、主程序流程图



2、中断子程序流程图



四、测试方案与测试结果

(1) 测试结果(数据)

直线下给定值与实测值测试结果如下表所示：

直线一半（单位/cm）

设定	15	16	17	18	19	20	21	22	23
实际	15.1	16.2	17.4	18.2	19.1	20.3	21.2	22.5	23.3

设定	24	25	26	27	28	29	30		
实际	24.5	25.4	26.7	27.5	28.4	29.5	30.6		

圆半径（单位/cm）

设定	15	16	17	18	19	20	21	22	23	24
实际	15.5	16.5	17.6	18.4	19.3	20.3	21.4	22.5	23.3	24.4

设定	25	26	27	28	29	30	31	32	33	34
实际	25.5	26.4	27.3	28.4	29.4	30.5	30.8	32.4	33.2	34.6

角度（单位/度）

设定	0	10	20	30	40	50	60	70	80
实际	0.5	10.3	20.5	30.6	40.7	49.3	61	69	81.2

设定	90	100	110	120	130	140	150	160	170
实际	90.1	101.2	110.4	120.2	131.1	139.3	149.2	160.5	171

(2) 测试分析与结论

根据上述测试数据：

- 1、直线较好的拟合长度
- 2、圆的半径稍大但在误差允许范围内
- 3、角度误差虽有点偏差但几乎在 1 度范围以内

综上所述，本设计达到设计要求。

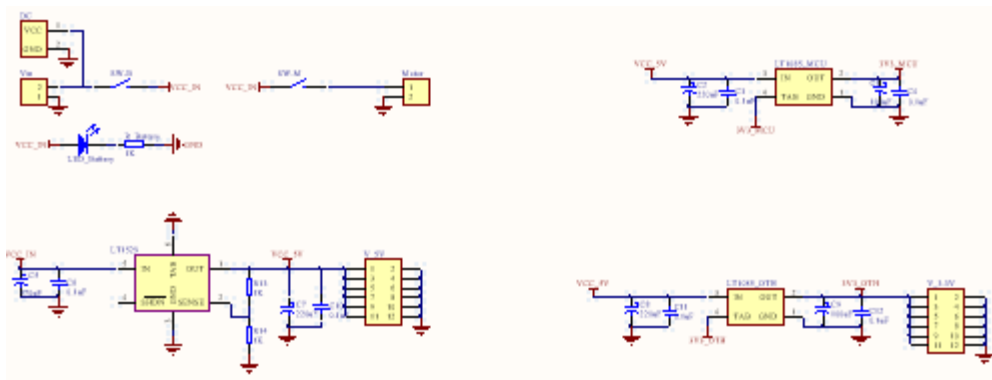
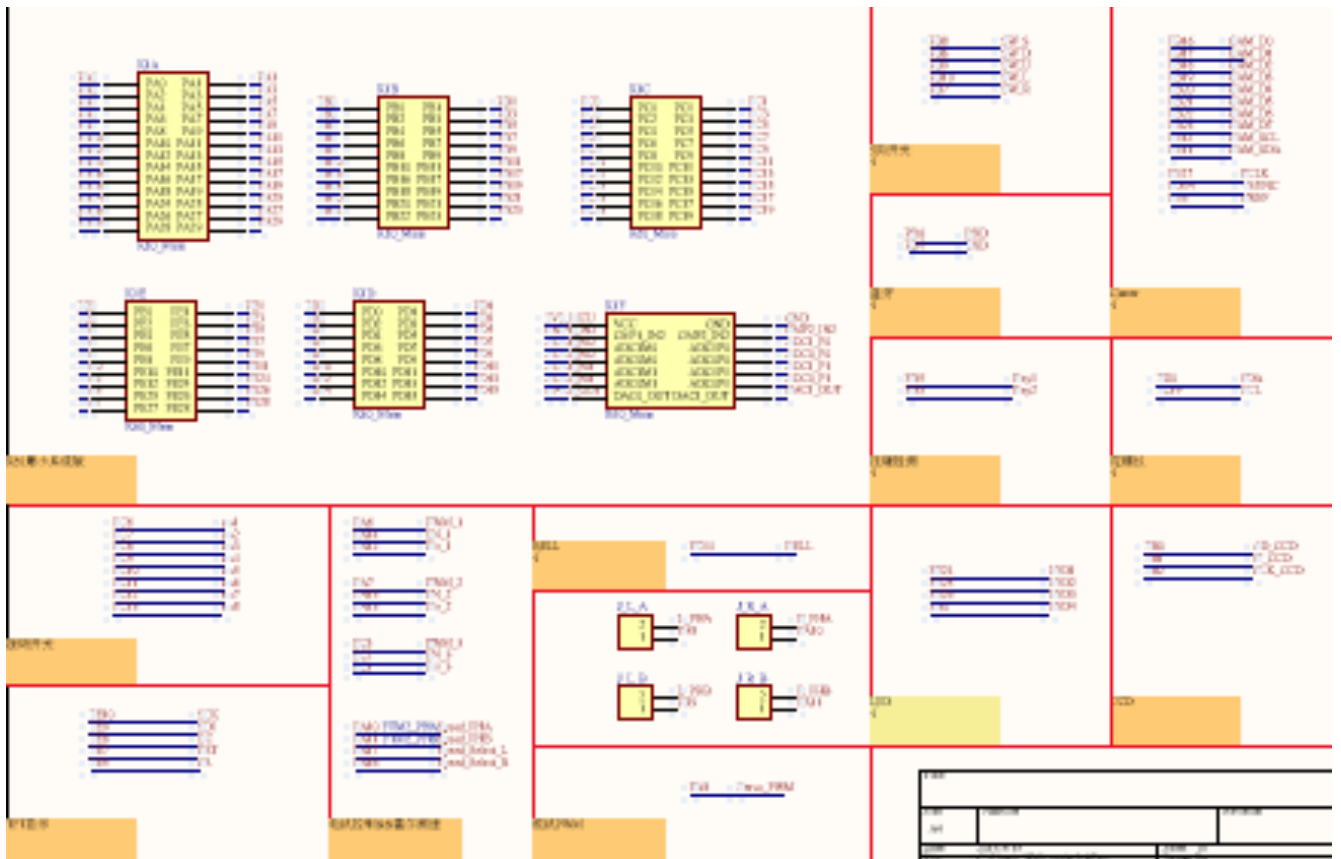
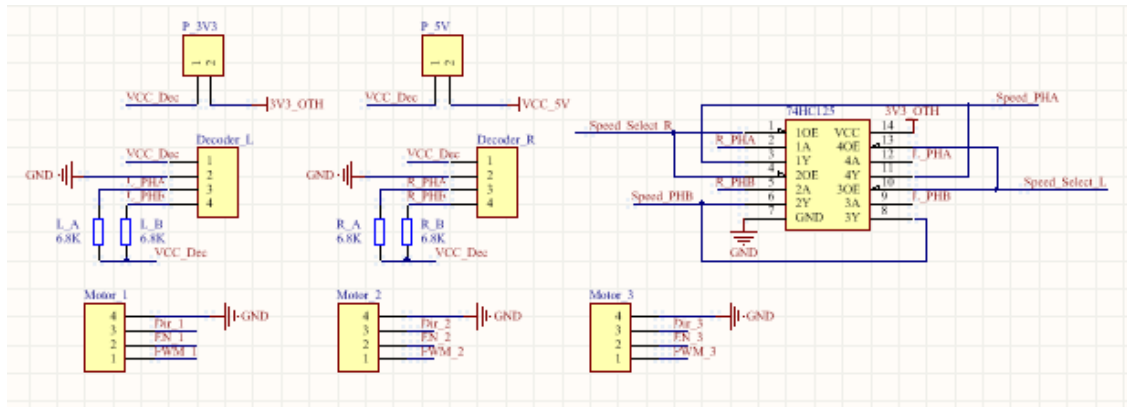
五、结论与心得

本次比赛，从硬件搭设到软件控制，队友一起协力完成。虽然中途硬件问题的影响一直进展的断断续续，但是我们并没有放弃，减少了休息，努力达到最好，每一次的参数调试都是思考后的结果。作品完成，结果已不重要，重要的是从准备到比赛结束一个多月以来我们积累到的各种知识以及实际调试中的经验将化为宝贵的财富受益终生。

六、参考文献

- [1] 谭浩强. C 语言程序设计[M]. 北京:清华大学出版社, 2012
- [2] 康光华. 模拟电子与线路[M]. 北京:华中科技大学出版社, 2012

附录 1: 电路原理图



附录 2：源程序

闭环部分代码：

```
void PID_aveY(float Aim1)
{

    error0_Y = Aim1 - angleY;
    error1_Y = angleY_his - angleY;
    angleY_his = angleY;
    error2_Y += I2*error0_Y;
    PWMC= (int)(P2* error0_Y+error2_Y+D2*error1_Y);
    if(PWMC>35000)
        PWMC = 35000;
    if(PWMC<-35000)
        PWMC = -35000;

    PWMA =- PWMC;
    if(mode3_flag==1)
    {
        //为 Y 正方向增加补偿值
        if(angleX<0&&angleY>0)
            PWMA+=pa;
        //为 Y 负方向增加补偿值
        if(PWMC>0)
            PWMC+=pc;
        //为 BC 方向增加补偿值
        if(PWMD<0)                                //&&PWMC>0
        {
            PWMC+=pbc;
            PWMB+=pbc;
        }
    }
    Speed();
}
```