

北航实验报告封面(共 8 篇)

北航惯性导航综合实验一实验报告

实验一

陀螺仪关键参数测试与分析实验 加速度计关键参数测试与分析实验

二零一三年五月十二日

实验一 陀螺仪关键参数测试与分析实验

一、 实验目的

通过在速率转台上的测试实验，增强动手能力和对惯性测试设备的感性认识；通过对陀螺仪测试数据的分析，对陀螺漂移等参数的物理意义有清晰的认识，同时为在实际工程中应用陀螺仪和对陀螺仪进行误差建模与补偿奠定基础。

二、 实验内容

利用单轴速率转台，进行陀螺仪标度因数测试、零偏测试、零偏重复性测试、零漂测试实验和陀螺仪标度因数与零偏建模、误差补偿实验。 三、 实验系统组成

单轴速率转台、MEMS 陀螺仪（或光纤陀螺仪）、稳压电源、数据采集系统与分析系统。 四、 实验原理

1. 陀螺仪原理

陀螺仪是角速率传感器，用来测量载体相对惯性空间的角速度，通常输出与角速率对应的电压信号。也有的陀螺输出频率信号（如激光陀螺）和数字信号（把模拟电压数字化）。以电压表示的陀螺

输出信号可表示为：

$$U_G = U_G(0) + k_G \omega + k_G f_G(a) + \Delta U_G$$

(1-1)

式中 $f_G(a)$ 是与比力有关的陀螺输出误差项，反映了陀螺输出受比力的影响，本实验不考虑此项误差。因此，式（1-1）简化为

$$U_G = U_G(0) + k_G \omega + \Delta U_G$$

(1-2)

由（1-2）式得陀螺输出值所对应的角速度测量值：

测量值 ω_m

$$\omega_m = \frac{U_G - U_G(0)}{k_G}$$

(1-3)

k_G

对于数字输出的陀螺仪，传感器内部已经利用标度因数对陀螺仪模拟输出进行了量化，直接输出角速度值，即：

测量值 ω_m 与真值 ω

(1-4)

$U_G(0)$ 是陀螺仪的零偏，物理意义是输入角速度为零时，陀螺仪输出值所对

应的角速度。且

$$U_G(0) = k_G \omega_0 \quad (1-5)$$

测量精度受陀螺仪标度因数 k_G 、随机漂移 ΔU_G 、陀螺输出信号 U_G 的检测精

度和 $UG(0)$ 的影响。通常 kG 和 $UG(0)$ 表现为有规律性，可通过建模与补偿方法消除， ωG 表现为随机特性，可通过信号滤波方法抵制。因此，准确标定 kG 和

$UG(0)$ 是实现角速度准确测量的基础。

五、陀螺仪测试实验步骤

1) 标度因数和零偏测试实验

- a. 接通电源，预热一定时间；
- b. 陀螺仪工作稳定后，测量静止情况下陀螺仪输出并保存数据；
- c. 转台正转，测试陀螺仪输出，停转；转台反转，测试陀螺仪输出，停转。在正转和反转时测试陀螺仪输出量，并分别保存数据；
- d. 改变转台输入角速率重复步骤 c，正负角速率的速率档分别不少于 5 个（按军标要求是 11 个）；
- e. 转速结束后，当转台静止时，采集陀螺仪输出数据，并保存。
- f. 根据最小二乘法公式

1

$i, j = 1, 2, \dots, M$

M

2

i, j

M

F

i, j

$j_1 M$

j_2

M

j

KG

1

$ij_1 M j_2 j_1$

(1-6)

$1FG$

MKG

F_j

Mj_1

M

j_1

M

ij

(1-7)

计算陀螺标度因数 and 零偏。 2) 零漂测试 (零偏稳定性)

在静止下采集陀螺仪数据, 并由测试数计算陀螺仪零偏稳定性。
军标中通常的测试时间是 1 小时, 并对所采集的数据进行 1 秒、
10 秒及 100 秒等不同时间的平滑。本实验中可采集数据 10 分钟

左右，并分别进行 1 秒、10 秒及 100 秒平滑。

按如下公式

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n \sigma_i^2$$

$$B_s = F_i \cdot F_{??} \cdot K_G \cdot n \cdot l_i \cdot 1??$$

$$l_2$$

$$(1-8)$$

计算陀螺仪零偏稳定性，并进行比较。 3) 零偏重复性测试

a. 令转台某角速度 $200^\circ/\text{s}$ 下进行正转，转速平稳后，采集陀螺输出数据，并保存。

b. 令转台某角速度 $-200^\circ/\text{s}$ 下进行反转，转速平稳后，采集陀螺输出数据，并保存。

c. 按计算陀螺零偏；

d. 关掉陀螺电源，并重新启动，重复步骤 a、b； e. 重复步骤 d 进行 3-5 次，共得到陀螺零偏 5-7 个； f. 对 5-7 个陀螺零偏按下式 (1-9)

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n \sigma_i^2 \quad (1-9)$$

$$Q_{li1}$$

$$l_2$$

求均方差，得零偏重复性指标。

六、 实验结果

1. 数据处理

将原始数据剔除后绘图如下

2. 计算陀螺标度因数和零偏

根据陀螺在 $10^{\circ}/s$, $20^{\circ}/s$, $40^{\circ}/s$, $60^{\circ}/s$, $80^{\circ}/s$ 角速率下正反转的输出, 分别求得正转下陀螺的标度因数和零偏, 及反转下陀螺的标度因数和零偏, 然后求的均值。 $KG= 0.9901$
 $FG0= 0.0358$

3. 零偏稳定性对所采集的数据进行 1 秒、10 秒及 100 秒等不同时间的平滑, 如下图。

篇二：北航计软实验报告实验一
实验报告

班
学
姓

成 级号名 绩 实验名称

篇三：北航 verilog 实验报告
北京航空航天大学
电子电路设计数字部分实验报告

实 验 一	简 单 组 合 逻 辑 设
计	
.....	2

实验二 简单分频时序逻辑电路的设计

一. 实验目的：1. 掌握最基本组合逻辑电路的实现方法。 3

2 . 学习时序电路测试模块的编写。

3 . 学习综合和不同层次的仿真。

3

实验三 利用条件语句实现计数分频时序电路

实验四 阻塞赋值与非阻塞赋值的区别

实验五 用 always 块实现较复杂的组合逻辑：

实验六 在 Verilog HDL 中使用函数

实验七 在 Verilog HDL 中使用任务 (task) 14

实验八 利用有限状态机进行时序逻辑的设计

实验九 楼梯

灯	
.....	19

实 验 思 考 与 总 结	
.....	29

学 院： 学 号： 姓 名：

实验一 简单组合逻辑设计

一. 实验目的：

1. 掌握基本组合逻辑电路的实现方法。
- 2 初步了解两种基本组合逻辑电路的生成方法。
- 3 学习测试模块的编写。
- 4 通过综合和布局布线了解不同层次仿真的物理意义。

二. 实验设备：

安装 Modelsim-6.5c 的 PC 机。

三. 实验内容：

描述一个可综合的数据比较器，比较数据 a 、 b 的大小，若相同，则给出结果 1，否则给出结果 0

四. 综合仿真结果

实验二 简单分频时序逻辑电路的设计

- 一. 实验目的：
1. 掌握最基本组合逻辑电路的实现方法。
 - 2 学习时序电路测试模块的编写。
 - 3 学习综合和不同层次的仿真。

二．实验设备：

安装 Modelsim-6.5c 的 PC 机。

三．实验内容：

用 always 块和 @(posedge clk) 或 @(negedge clk) 的结构表述一个 1/2 分频器的可综合模型，观察时序仿真结果

四．实验代码

```
module half_clk(reset,clk_in,clk_out);  
input clk_in,reset;  
output clk_out;  
reg clk_out;  
  
    always@(posedge clk_in)  
begin  
if(!reset)  
clk_out=0;  
else  
clk_out=~clk_out;  
end  
endmodule  
  
`timescale 1ns/100ps  
`define clk_cycle 50  
  
module top;  
reg clk,reset;  
wire clk_out;
```

```

always #`clk_cycle clk=~clk;

initial

begin

clk=0;

reset=-1;

#10 reset=0;

#110 reset=1;

#100000 $stop;

end

half_clk m0(.reset(reset),.clk_in(clk),.clk_out(clk_out));

endmodule

```

实验三 利用条件语句实现计数分频时序电路

一. 实验目的:

1. 掌握条件语句在简单时序模块设计中的使用。

篇四：2014 级北航单片机实验报告

北京航空航天大学研究生课程考核记录

2014-2015 学年 第一学期

学号姓名 成绩

课程名称： 《 单片机系统实验 》

论文题目： 实验报告

任课教师评语：

一、实验目的 考核日期： 年月日

1. 了解一款 32 位单片机（STM32 系列）原理及其应用，熟悉单片机的资源，掌握单片机的最小系统设计及扩展技术，掌握单片机的编程语言。

2. 通过本实验了解 LCD 液晶工作原理，能通过编程操作液晶的显示。

二、实验设备

STM32 实验系统一套，PC 机一台。

三、实验原理

1. I/O 口及定时器实验：STM32 的 GPIO 口控制 4 个发光二极管，了解其硬件连接方式，学会使用 STM32 的一个定时器，掌握对定时器计时方式的编程。编写程序循环点亮 4 个发光二极管，控制点亮时间为 1 秒钟闪烁。

2. LCD 实验：掌握 STM32 单片机与液晶之间的硬件连接方式，单片机如何驱动液晶进行显示。

四、内容与步骤

1. 学会使用 IAR 的编译链接调试环境，熟悉 IAR 中有关 STM32 使用到的库，并能顺利建立包含各种库文件的工程。

2. I/O 口及定时器实验：循环定时（用定时器做）点亮 4 个灯，即每 1 秒闪烁点亮一个灯，循环往复（或叫跑马灯实验）。

3. LCD 实验：通过自行编写库文件和了解液晶显示字库，能在

五、实验程序

实验一：I/O 口及定时器实验

STM32的 GPIO口控制 4 个发光二极管循环点亮，用定时器计时设置 LED灯的点亮时间为 1 秒。

开发板原理图中 LED模块的原理图如图 1 所示，与 LED灯相连的分别是 PD8~PD11 四个 I/O 接口，且 4 个 LED灯共阴极，即在 LED灯对应的 I/O 口输出高电平时可点亮该 LED灯。

图 1 LED灯硬件原理图

实验中选择定时器 TIM2，系统时钟是 72MHz，为了实现预定的定时时间，需要配置的关键参数是时钟预分频数 TIM_Prescaler 和自动重装载寄存器周期的值 TIM_Period。实验中需要定时 1 秒，则设置定时器 TIM2 为 7200 分频，即 TIM_Prescaler=7199；设置 TIM_Period=9999。其计算过程如下：

$$1 \text{ 秒} = \frac{1 + \text{TIM_Prescaler}}{72\text{M}} \times (1 + \text{TIM_Period}) \times \frac{1}{72\text{M}} = \frac{10000}{72\text{M}}$$

程序代码如下：

```
#include stm32f10x.h
```

```
#define _TIM
```

```
#define _TIM2
```

```
/*配置系统时钟，设置系统时钟为 72MHz*/
```

```
void RCC_Configuration()
```

```

    ErrorStatus HSEStartUpStatus;    // 义枚举类型变量

HSEStartUpStatus

    RCC_DeInit(); //复位系统时钟    RCC_HSEConfig(RCC_HSE_ON); //
开启 HSE    HSEStartUpStatus = RCC_WaitForHSEStartUp(); //等待
HSE起振    if(HSEStartUpStatus == SUCCESS)
    {

    FLASH_PrefetchBufferCmd(FLASH_PrefetchBuffer_Enable);
    FLASH_SetLatency(FLASH_Latency_2);
    RCC_HCLKConfig(RCC_SYSCLK_Div1);
    //选择 HCLK时钟源为 SYSCLK分频
    RCC_PCLK2Config(RCC_HCLK_Div1);
    //选择 PCLK2的时钟源为 HCLK1分频
    RCC_PCLK1Config(RCC_HCLK_Div2);
    //选择 PCLK1的时钟源为 HCLK2分频
    RCC_PLLConfig(RCC_PLLSource_HSE_Div1, RCC_PLLMul_9); //选
择 PLL时钟源为 HSE1分频，9 倍频
    RCC_PLLCmd(ENABLE); //使能 PLL
    while(RCC_GetFlagStatus(RCC_FLAG_PLLRDY) == RESET);
    //等待 PLL输出稳定
    RCC_SYSCLKConfig(RCC_SYSCLKSource_PLLCLK);
    //系统时钟为 PLL

```



```

while(RCC_GetSYSCLKSource()!= 0x08); // PLL为系统时钟
源 }

RCC_APB2PeriphClockCmd( RCC_APB2Periph_USART1 |
RCC_APB2Periph_GPIOA |
RCC_APB2Periph_GPIOC|RCC_APB2Periph_GPIOB|RCC_APB2Periph
_GPIOD|
RCC_APB2Periph_GPIOE, ENABLE );
RCC_APB1PeriphClockCmd(RCC_APB1Periph_SPI2,ENABLE);
}

/*GPIOD初始化设置*/
void GPIOinit()
{
GPIO_InitTypeDef GPIO_InitStructure;
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_All; //选中所有的引
脚 GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
//I/O 口 50MHz输出速度
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP;
//设置 I/O 口推挽输出
GPIO_Init(GPIOD&GPIO_InitStructure);// 初始化外设 D 端
口 }

/*初始化定时器*/
void time_init(void)

```

```

//          72MHz : 7200 分频   1 秒中断一次，向上计数
TIM_TimeBaseInitTypeDef TIM_TimeBaseStructure;
NVIC_InitTypeDef NVIC_InitStructure;
RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM2 , ENABLE);
TIM_TimeBaseStructure.TIM_Period = 9999; // 自动重装载寄存器周期的值
TIM_TimeBaseStructure.TIM_Prescaler = 7200-1; // 时钟预分频数
TIM_TimeBaseStructure.TIM_ClockDivision = 0; // 时间分割值
TIM_TimeBaseStructure.TIM_CounterMode = TIM_CounterMode_Up; // 向上计数
TIM_TimeBaseInit(TIM2,&TIM_TimeBaseStructure); // 初始化定时器
TIM_ITConfig(TIM2,TIM_IT_Update,ENABLE); // 开定时器中断
TIM_Cmd(TIM2,ENABLE); // 使能定时器 // 使能 TIM2 中断
NVIC_PriorityGroupConfig(NVIC_PriorityGroup_1); // 配置中断分组
NVIC_InitStructure.NVIC_IRQChannel = TIM2_IRQn; // 配置中断线路
NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0;
// 配置抢占优先级为 0
NVIC_InitStructure.NVIC_IRQChannelSubPriority = 0;
// 配置从占优先级为 0
NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE; // 打开 NVIC 使能
NVIC_Init(&NVIC_InitStructure); // 初始化 }

```

```

int i=0;

void main()
{
    RCC_Configuration(); //配置系统时钟
    GPIOinit(); //GPIOD初始化
    time_init(); // 初始化定时器
    while(1)
    {
        switch(i)
        {
            case 0 :GPIO_Write(GPIOD,0x0100);break;//PD8 置 “1”，点亮第
一个灯 case 1 :GPIO_Write(GPIOD,0x0200);break;//PD9 置 “1”，点
亮第二个灯 case 2 :GPIO_Write(GPIOD,0x0400);break; //PD10 置
“1”，点亮第三个灯 case 3 :GPIO_Write(GPIOD,0x0800);break;
//PD11 置 “1”，点亮第四个灯  }
        }
    }
}

```

中断服务函数：

```

void TIM2_IRQHandler(void)
{
    if (TIM_GetITStatus(TIM2,TIM_IT_Update) != RESET)
    {

```



```
TIM_ClearITPendingBit(TIM2,TIM_IT_Update);  
  
i++;  
  
if (4 == i)  
  
i = 0;  
  
}  
  
}
```

运行上述程序可以循环点亮 4 个 LED 灯，时间间隔为 1 秒。

实验二：LCD 实验

借助字模提取软件提取字模点阵数据并存储。LCD 显示采用 SSD1298 驱动器显示窗口设置和地址自增的功能实现快速显示，首先判断坐标位置是否超出边界，然后设置显示窗口大小，设置起始点，发送写入 GRAM 命令，调用汉字显示的点阵数据，遍历并判断每一位是 1 或者 0，则对应的点是显示画笔颜色还是背景颜色，字模读取顺序选择从左到右，再从上到下，最终在液晶显示屏上显示“北京航空航天大学”。

篇五：北航自控实验实验报告模板(1)

成绩

北 京 航 空 航 天 大 学

自动控制原理实验报告

学 院 专业方向

班 级 学 号 学生姓名指导教师

自动控制与测试教学实验中心

实验一 一、二阶系统的电子模拟及时域响应的动态测试

实验时间 2014. 10. 28 实验编号 35 同组同学 无 一、实验目的

了解一、二阶系统阶跃响应及其性能指标与系统参数之间的关系； 学习在电子模拟机上建立典型环节系统模型的方法； 学习阶跃响应的测试方法。

二、实验内容

建立一阶系统的电子模型，观测并记录在不同时间常数 T 时的阶跃响应曲线，并测定其过渡过程时间 T_s ；

建立二阶系统的电子模型，观测并记录在不同阻尼比时的阶跃响应曲线，并测定其超调量及过渡过程时间。

三、实验原理

1.一阶系统

系统传递函数为： $\Phi(s)$ =模拟运算电路如图 1- 1所示：

图 1- 1

由图 1-1得

在实验当中始终取 $R_2 = R_1$ ，则 $K=1$ ， $T = R_2C$ 取不同的时间常数 T 分别为： 0.25、 0.5、 1 。

2二阶系统

其传递函数为：令 $\omega_n=1$ 弧度/秒，则系统结构如图 1-2所示：

自动控制与测试教学实验中心 1

图 1-2

根据结构图，建立的二阶系统模拟线路如图 1-3所示：

图 1-3

取 $R_2C_1=1$, $R_3C_2=1$, 则

ζ 取不同的值 $\zeta=0.25$, $\zeta=0.5$, $\zeta=1$ 实验电路图 一阶:

及

二阶:

自动控制与测试教学实验中心

2

四、实验步骤

1熟悉 HHMN-1型电子模拟机的使用方法,将各运算放大器接成比例器,通电调零; 2 断开电源,按照实验说明书上的条件和要求,计算电阻和电容的取值,按照模拟线路图搭接线路,不用的运算放大器接成比例器;

3 将 D/A1 与系统输入端 U_i 连接,将 A/D1 与系统输出端 U_o 连接;

4在 windows XP 桌面用鼠标双击 MATLAB图标后进入,在命令行键入 autolab , 进入实验软件系统;

5在系统菜单中选择实验项目,选择实验一,在窗口左侧选择“实验模型”;

6观测实验结果,记录实验数据,绘制实验结果图,填写实验数据表格,完成实验报告。

五、实验结果

1、一阶系统:

实验结果图：

$T=0.25$:

$T=0.5$:

自动控制与测试教学实验中心 3

$T=1$:

自动控制与测试教学实验中心

4

篇六：北航数图实验报告三 图像复原实验

北 京 航 空 航 天 大 学

数字图像处理实验报告

学 院

专业方向

班 级

学 号

学生姓名

指导教师

实验三 图像复原实验

1. 实验目的

利用反向滤波和维纳滤波进行降质图像复原，比较不同参数选择对复原结果的影响。

2 实验内容

(1) 利用反向滤波方法进行图像复原；

(2) 利用维纳滤波方法进行图像复原。

3. 实验要求

(1) 输入图像采用实验 1 所获取的图像，对输入图像采用运动降质模型，如下式所示

错误！不能通过编辑域代码创建对象。

与降质图像相关的参数是：错误！不能通过编辑域代码创建对象。；

(2) 对每一种方法通过计算复原出来的图像的峰值信噪比，进行最优参数的选择，包括反向滤波方法中进行复原的区域半径错误！不能通过编辑域代码创建对象。、维纳方法中的噪声对信号的频谱密度比值 K ；

(3) 将降质图像和利用最优参数恢复后的图像同时显示出来，以便比较。

4. 程序代码

1.运动降质的源代码

```
I=imread('saber.bmp');  
I=rgb2gray(I);  
F=fft2(I);  
F=fftshift(F);  
for m=-((size(F,1)+1)/2):(size(F,1)+1)/2  
for n=-size(F,2)/2:size(F,2)/2
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/486231022120010232>