

控制理论实验指导书

韩涛

应用电子与自动化实验教学中心

目录

绪论	3
第一部分 控制系统动态特性的时域及频域测试	6
§1—1 控制系统动态特性的时域测试	6
§1—2 控制系统动态特性的频域测试	8
控制系统的电子模拟实验	13
§2—1 电子模拟实验的基本部件	13
§2—3 典型环节的模拟及参数测试	17
§2—4 控制系统的模拟	21
随动系统	23
§3—1 概述	23
§3—2 随动系统的结构原理	23
§3—3 随动系统的主要部件及其传递函数	24
实验内容	40
实验一 典型环节的模拟实验	40
实验二 一阶系统的时域响应及参数测定	45
实验三 二阶系统的瞬态响应分析	47
实验四 三阶系统的瞬态响应及稳定性分析	50
实验五 频率特性的测量	53
实验六 线性定常系统的串联校正	57
第二部分 仿真环境简明介绍	62
§1 MATLAB	62
§2 SIMULINK 简介	83
§3 SIMCAP 模型库概述	98
§4 PYTHON-CONTROL 使用介绍	110
第三部分 数字仿真实验	127
实验六 控制系统的模型转换	127
实验七 控制系统的时域分析	129
实验八 控制系统的根轨迹分析	130
实验九 控制系统的频域分析	131

实验十 控制系统的极点配置	1 3 3
附录	1 3 4
DS1102E 数字示波器简介	1 3 4
DG1022U 双通道函数/任意波形发生器简介	1 4 7

绪论

随着现代科学技术的迅速发展，自动控制技术在许多工程领域（如电力工程、机械工程、化工工程、生物医学工程、航空航天以及交通运输等）有着广泛的应用。随着对自动控制系统性能要求不断提高，控制系统的结构也愈趋复杂，为了处理新的问题，需要新的分析方法。以频率法或根轨迹法为基础处理单输入单输出系统的问题，称为经典控制理论。随着科学技术的发展，为了适应各种更广泛、更复杂的控制要求，例如多输入多输出系统、时变系统等等，一种以状态空间为基础的现代控制理论迅速发展起来。应用现代控制理论，设计者可以按照所要求的各种性能指标，设计最优化的系统，即选用最合适的控制规律。

控制理论所要处理的问题很多，这里只介绍最基本的三类问题：

第一类是研究系统的稳定性。所谓稳定性，就是指在受到扰动作用之后，系统的被控制量虽然偏离了原来的平衡状态，但当扰动撤离，经过一段时间后，如果系统仍然能回到原有的平衡状态，则称系统是稳定的。一个稳定的系统，当其内部参数稍有变化或者初始条件改变时，仍能正常地进行工作。考虑到实际系统在工作过程中的环境和参数的变化，因此在设计实际系统时不仅要求能稳定，还要求留有一定的稳定裕量。

第二类是研究系统的准确度(或称稳态精度)。系统的准确度是指在调整过程结束后，输出量与参考输入量之间的偏差。这种偏差越小，表示系统的输出跟随参考输入的精度越高。通常用阶跃、斜坡或抛物线输入系统后的稳态误差来评定。

第三类是研究系统的动态性能指标。系统的动态性能指标有时域性能指标和频域性能指标两类。

根据被控对象的不同，各种系统对上述三方面性能要求的侧重点也有所不同。例如随动系统对响应速度和稳态精度的要求较高，而恒值控制系统一般侧重于稳定性能和抗扰动的能力。在同一个系统中，三方面的性能要求又通常是相互制约的。

随着对自动控制系统性能要求不断提高，控制系统的结构也愈趋复杂，为了处理新的问题，需要新的分析方法。以频率法或根轨迹法为基础处理单输入单输出系统的问题，称为经典控制理论。随着科学技术的发展，为了适应各种更广泛、更复杂的控制要求，例如多输入多输出系统、



图 0-1 控制系统设计

计流程图时变系统等等，一种以状态空间为基础的现代控制理论迅速发展起来。应用现代控制理论，设计者可以按照所要求的各种性能指标，设计最优化的系统，即选用最合适的控制规律。

设计一个控制系统，可用图 0-1 的流程图来表示。当然，在具体的设计过程中，图中的七个步骤不是截然分开的，各个方框之间要经过多次反复实验，使它们之间交叉联成一个整体。在设计和分析控制系统时，既要强调理论基础，又要重视实验技术。控制理论实验的主要任务是流程图中的中间三个步骤，即从进行性能分析到仿真实验。

实验研究的方法有两种：一种是在真实系统(称原型)上进行，另一种是在模型上进行。对于比较简单的被控对象，可以在实际系统上进行试验和调整，以获得较好的性能指标。但是在生产过程中，大部分的被控对象是比较复杂的，如火箭发射、反应堆控制、轧钢机调速系统、电力系统等等，考虑到安全性、经济性以及进行实验研究的可能性诸原因，在实际系统上进行实验，通常是不允许的。这时，就需要把实际系统建立成数学模型来进行研究，然后把对模型实验研究的结果应用到实际系统中去。这种方法叫做模拟或仿真研究，简称“仿真”。

物理仿真就是用缩小(或放大)了的尺寸(或容量)，制作与系统原型相同的模型，模型的变量与真实系统的变量完全一致。例如，要研究一个大功率的晶闸管直流调速系统。我们可用小容量的电动机、发电机(作负载用)及小容量的晶闸管整流装置组成一个物理仿真系统。物理仿真造价较高，而且要作大量的安装、调整等工作，因而准备的周期较长。

数字仿真是将实际系统的运动规律用数学方程(一组微分方程或差分方程)来描述，然后用数字计算机来解数学方程。在数学仿真中，由于仿真模型和真实系统具有相同的数学模型。因此就动态性能而言，对模型的研究，就可以代替对真实系统的研究。

数字仿真比物理仿真具有更广泛的用途，它可以对物理性质截然不同的许多控制系统进行研究。数字仿真的准备工作比物理仿真所需准备的工作量要小，周期也要短，所化的费用也少，试验数据的处理也简单，这是数字仿真比物理仿真发展更迅速的原因。

控制理论实验分为两大部分：

1. 物理仿真。包括利用运算放大器的基本特性(开环增益高、输入阻抗大、输出阻抗小等)，设置不同的输入网络和反馈网络来实现各种典型环节，再由各种典型环节按实验要求可连接起来构成整个系统的电子模拟装置以及小功率模拟随动系统。

2. 数字仿真。主要包括基于 MATLAB 语言的控制系统分析和设计以及 Simulink 仿真。

第一部分 控制系统动态特性的时域及频域测试

§ 1—1 控制系统动态特性的时域测试

控制系统的动态特性是指系统在动态过程(过渡过程)中输出量对于输入量的时间函数关系。由于大多数控制系统是以时间作为独立变量的,因此系统的动态特性往往用时域响应来描述。一般而言,只要在阶跃信号输入下系统的时域响应能符合设计要求,则在其它任何信号输入下,系统的动态性能指标能满足要求。实验获得阶跃响应的方法很多。其中最简单的一种就是直接给于系统一个阶跃输入量,并同时用测试仪器(慢扫描示波器、光线示波器或函数记录仪等)测量其输出量。假如测得实际系统的阶跃响应如图 1-1-1 所示,则系统的动态性能可用该图中所示的性能指标来表征(图中取初始条件为零)。

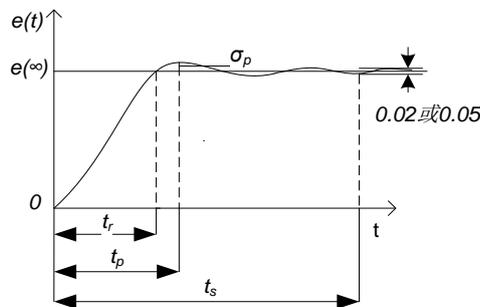


图 1-1-1 阶跃响应的性能指标

一、上升时间 t_r 。对于过阻尼系统它是响应从终值的 10%上升到 90%所需要的时间,图 1-1-1 表示欠阻尼系统。

二、峰值时间 t_p 。当响应曲线到达第一个峰值所需的时间。

三、超调量 $\sigma_p\%$ 。响应曲线超过阶跃输入的最大偏离量。通常表示为阶跃响应终值的百分数,即 $\sigma_p\% = (C(t_p) - C(\infty)/C(\infty)) * 100\%$ 。

四、调整时间 t_s 。响应曲线衰减到并停留在终值的某一规定的误差带(2%或 5%)内所需的时间。

五、振荡次数 N 。在调整时间内,响应曲线穿越其终值 $C(\infty)$ 次数的一半。

在分析和设计控制系统时,上述性能指标不是全部都要采用的。应根据系统的使用条件和实际情况,只对其中几个认为重要的性能指标提出要求。

对线性系统可用时域法研究控制系统的动态特性。先对系统加入一个输入信号后，测定系统的输出响应，然后根据此响应曲线，求出系统的数学方程。所加的输入信号一般为阶跃信号，由于时域法能比较直观地反映系统的动态特性，无需中间转换，实验原理也最简单，因此这是一种基本的测定系统动态特性的方法，这种测试方法的原理图如图 1-1-2 所示。

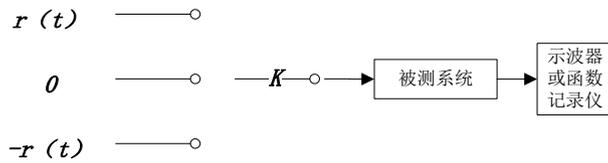


图 1-1-2 阶跃响应测试原理图

测试时要注意以下情况：

1. 阶跃信号幅值的大小选择应适当考虑。过大会使系统动态特性的非线性因素增大，使线性系统变成非线性系统；过小也会使系统信噪比降低并且输出响应曲线不可能清楚显示或记录下来。
2. 在阶跃信号施加前，应使系统保持相当长的稳定运行。在相同的阶跃幅值输入下，检验被测的正向和反向阶跃响应曲线，如果两者波形和幅值基本一样，即表明系统确为线性；否则须按非线性系统处理。
3. 恰当选择示波器或函数记录仪的“扫描时间”旋钮，使阶跃响应曲线的波形完整、清楚地显示或记录下来。

§ 1—2 控制系统动态特性的频域测试

系统动态性能指标可用时域法来分析，也可用系统的频率特性来评价。前者的优点是直观逼真，后者对工程分析和设计比较方便。可有效地利用频率特性曲线，而不必求解复杂的解析式，这一点对于无法取得数学模型的某些复杂对象尤为重要。

系统的频率特性测试要比时域响应测试复杂，但由于测试频率特性时，被测系统施加一种稳态正弦信号，系统处在稳态，外来随机干扰对测试结果的影响比测试时域响应时小得多，因此测量准确度较高。此外，对于最小相位系统，在未知系统传递函数的情况下，则可对感兴趣的频率范围内，通过实验测试系统的对数频率特性，并对此作出对数频率特性曲线，即可求出具有一定精度的系统开环传递函数。测试控制系统频率特性有很多方法，下面就几种常用的方法作些简要说明。

一、输入输出曲线直接记录法

测试框图如图 1-2-1 所示。正弦信号发生器固定一个频率 ω ，待输出稳定后，即可以在双线示波器或双线记录仪上获得一组输入输出曲线，如图 1-2-1 (b) 所示。其中输出曲线常常夹杂着干扰，因而已不是光滑的正弦曲线了，但是峰值比还是容易确定的。

测量幅频特性时，一般为了读取方便，输入和输出的幅值均取其峰峰值，即

$$A(\omega_i) = 2Y_m(\omega_i)/2X_m(\omega_i)$$

测量相频特性时，可测量输入与输出峰值间的距离 d_0 ，并量出输入曲线前

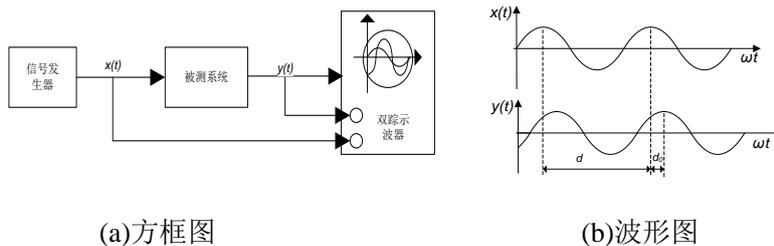


图 1-2-1 直接记录法测试频率特性的原理图

后两个峰间的距离 d ，则可得到某一频率为 ω_i 时的相位差值

$$\varphi(\omega_i) = -\left(d_0 * \frac{360^\circ}{d}\right)$$

在上式中，若输出超前于输入，则 $\varphi > 0$ 符号应取正。这样，在测定的频率范围内逐渐改变输入频率 ω_i ，重复上述测量，就可得到一系列对应不同频率 ω_i 的幅值比和相位差的值，从而得到被测系统的幅频特性和相频特性曲线。

用直接记录法测量 $A(\omega)$ 和 $\varphi(\omega)$ 需要逐点测量和换算，使用较麻烦，测量精度较低，谐波与噪声抑制能力也较差，因此目前一般已不再采用。

二、李沙育图形法

将正弦信号作为输入信号，并和被测系统(或环节)的输出分别接到慢扫描示波器的 X 轴和 Y 轴上，如图 1-2-2 所示，就可以在示波器上显示李沙育图形。设被测系统的输入量 $x(t)$ 和输出量 $y(t)$ 的表达式分别为：

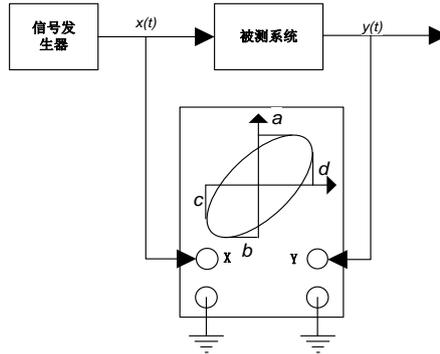


图 1-2-2 李沙育图形测试频率特

$$x(t) = X_m \sin \omega t$$

$$y(t) = Y_m \sin[\omega t + \varphi(\omega)]$$

则李沙育图形的产生如图 1-2-3 所示。若以 t 作为参变量，则随 t 的变化， $x(t)$ 和 $y(t)$ 所确定的点的轨迹，将在示波器屏幕上形成一条封闭的曲线(通常是一个椭圆)。这就是所谓的李沙育图形

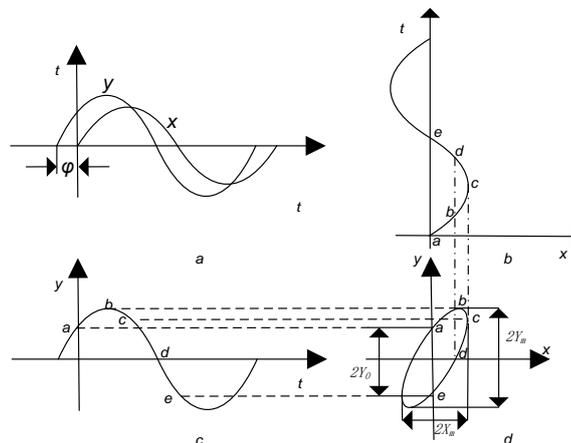


图 1-2-3 李沙育图形测试频率特性

(a)输入输出波形；(b)示波器 X 轴的输入波形；(c)示波器 Y 轴的输入波形；(d)李沙育图形

由图可知, $t = 0$ 时, $x(0) = 0$

$$y(0) = Y_m \sin(\varphi(\omega))$$

得:

$$\varphi(\omega) = \sin^{-1} 2Y_0/2Y_m \quad (1-2-1)$$

同理:

$$\varphi(\omega) = \sin^{-1} 2X_0/2X_m \quad (1-2-2)$$

其中: $2Y_0$ 为椭圆与Y轴交点之间的长度。

$2X_0$ 为椭圆与X轴交点之间的长度。

上式适用于椭圆长轴在第一、三象限;当椭圆长轴在第二、四象限时计算公式为

$$\varphi(\omega) = 180^\circ - \sin^{-1} 2Y_0/2Y_m \quad (1-2-3)$$

$$\varphi(\omega) = 180^\circ - \sin^{-1} 2X_0/2X_m \quad (1-2-4)$$

因此,在某频率范围内逐点改变频率 ω 重复测量,就可得到相频特性和幅频特性

$$\angle G(j\varphi) = \varphi(\omega) = \sin^{-1} 2Y_0/2Y_m$$

$$|G(j\varphi)| = 2Y_m/2X_m.$$

实际测量相频特性时,通常保持信号发生器的输出幅值不变.因此,采用(1-3-2)式和(1-3-4)式为好.这样既可减少实验工作量,也可提高测量精度.因为整个实验过程中 $2X$.只需测量一次即可。

相角 ϕ	超前		滞后	
	$0^\circ \sim 90^\circ$	$90^\circ \sim 180^\circ$	$0^\circ \sim 90^\circ$	$90^\circ \sim 180^\circ$
图形				
计算公式	$\varphi = \sin^{-1} \frac{2Y_0}{2Y_m}$	$\varphi = 180^\circ - \sin^{-1} \frac{2Y_0}{2Y_m}$	$\varphi = \sin^{-1} \frac{2Y_0}{2Y_m}$	$\varphi = 180^\circ - \sin^{-1} \frac{2Y_0}{2Y_m}$
光点转向	顺时针	顺时针	逆时针	逆时针

表 1-2-1

应当指出，系统输出大多是迟后于输入的，在这种情况下，示波器上的光点作逆时针运动，用上述公式计算的结果需加负号。如果光点是顺时针运动，则输出 $y(t)$ 超前于输入 $X(t)$ ，计算结果应为正。上述原理见表 1-2-1。

李沙育图形法对仪器要求不高，但所得的精度较低，特别在频率较高时，光点运动方向不易看出，这时只能按测试的数据的连续性和对测试系统(或环节)的初步了解来估算其符号。

三、补偿法

用补偿法测试控制系统频率特性的原理如图 1-2-4 所示。这是使用超低频系列仪器组合在一起测量被测系统的幅值比和相位差。幅频特性的测试较容易，在输入幅值给定的情况下，改变信号频率，断开开关 K ，用超低频示波器直接测量出系统输出的幅值，即可求出幅值比。相频特性是借助于超低频移相器从示波器显示的李沙育图形来测得对应的相角，当两个频率相同的正弦信号，同时从示波器的 X、Y 轴输入时，显然没有相位差，其图形一定是一条直线，根据这个原理，把被测系统的输出加到示波器的 Y 轴，而移相器的输出加到示波器 X 轴，由于两个信号有相位差，故示波器上呈现椭圆，只要调节移相器的相位旋钮，当示波器显示一直线时即停止调节。此时移相器上指示的相角值就是被测系统输出的相位差。这种测量方法，抗干扰能力差，因测量值取峰值，在被测系统中可能存在非线性因素，输出并不是正弦波，故测得的峰值不是基波的幅值，因此会造成误差。

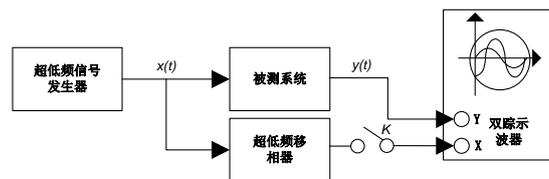


图 1-2-4 补偿法测试频率特性的原理图

四、相关测量法

现代的频率特性测试仪一般都采用相关测量原理来测试系统的动态特性，其测试频率特性的原理如图 1-2-5 所示。从图中可得知相关测量法是将被测系统的输出信号与参考信号 $\sin\omega t$ 和 $\cos\omega t$ 分别相乘，然后在基波的整数周期内积分并求其平均值，则可得到基波分量的实部和虚部，同时抑制直流分量、高次谐波和噪声。故测量精度高，目前应用十分广泛。

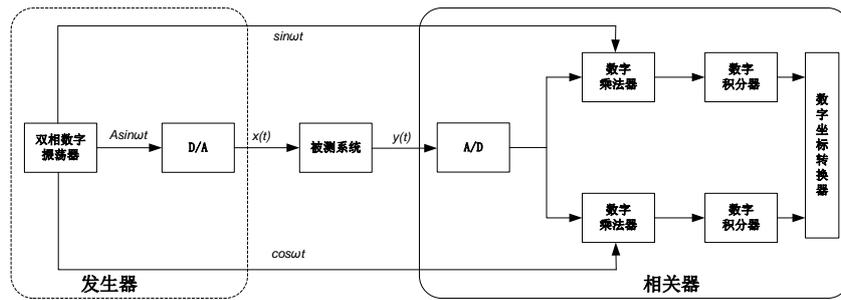


图 1-2-5 相关法测试系统频率特性的框图

在实际测试频率特性时要注意以下情况：

1. 由于被测系统具有某些非线性因素，必须适当选择输入正弦信号的幅度。如果输入信号太大，会引起输出信号饱和；输入信号太小，也会由于死区而引起误差。这可通过检查输出信号是否接近正弦波形来考虑输入信号的大小。
2. 试验前应先确定频带范围，求出最大频率 ω_{max} ，在已定的初始频率 ω_i 下，逐渐增大输入正弦信号的频率，直到输出幅值 $Y_m(\omega_{max})$ 只有 ω_i 幅值 $Y_m(\omega_i)$ 的 1/20~1/100。把 ω_i 到 ω_{max} 这一频带分成 5 至 10 段，确定 ω_1 、 ω_2 、 ω_3 、…然后把选好的频率依次逐个地加到被测系统上，记下所测试数据，得出被测系统的幅频特性和相频特性。
3. 若用超低频系列仪器测试频率特性，则应注意超低频示波器交流耦合的下限频率。如输入信号频率低于下限频率时，则必须用直流档输入。
4. 对被测系统进行理论分析时，总是认为信号源输出阻抗为零，测量仪器的输入阻抗为无穷大。但实际测试时，只有当被测系统的输入抗远大于信号发生器的输出阻抗及被测系的输出阻抗远小于测量仪器的辅入阻抗时，论分析才能接近于实测值。

控制系统的电子模拟实验

§ 2—1 电子模拟实验的基本部件

电子模拟装置的基本部件都是运算放大器，它被用来作为积分器、比例器、加法器、系数器等运算部件。这些运算部件均属于线性运算部件，运算放大器视为满足以下条件的为理想放大器：

1. 放大器的开环放大倍数为无限大；
2. 开环输入阻抗为无限大，输出阻抗等于零；
3. 通频带为无限大；
4. 输入与输出间呈线性特性。

由于运算放大器的通频带远远大于控制系统中各种典型环节的工作频段，上述条件 3 一般均能满足。当输入信号取得比较小时，运算放大器工作在线性区，因此可以满足条件 4。

下面介绍几种常用运算部件的线路图（输入信号使用反相输入端）。

一、比例器

比例器的线路图 2-1-1 所示，运算放大器的输入、输出关系为反馈阻抗和输入阻抗之比。在比例器中，输入量只有一个，而且输入阻抗和输出阻抗均为电阻，所以其输入电压和输出电压的关系

$$\frac{u_o}{u_i} = \frac{R_f}{R_i} \quad (2-1-1)$$

式 $k = \frac{R_f}{R_i}$ ——比例器的传递系数(又称放大倍数)。

改变 R_f 或 R_i 的电阻值便可以改变比例器放大倍数 K 。

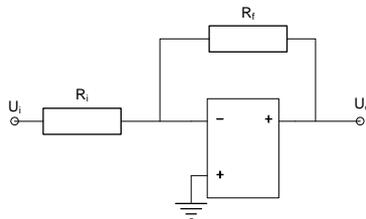
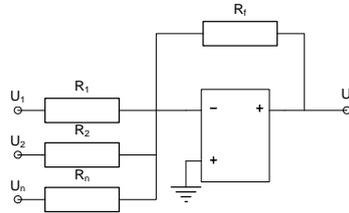


图 2-1-1 比例器的线路图

二、加法器

加法器是完成多个信号相加运算的部件。使用它可模拟各种信号的综合。

图 2-1-2 为加法器的线路图，加法器的输出电压和输入电压的关系为



2-1-2 加法器的线路图

$$u_0 = -\left(\frac{R_f}{R_1}u_1 + \frac{R_f}{R_2}u_2 + \frac{R_f}{R_3}u_3 + \cdots + \frac{R_f}{R_n}u_n\right) = \sum_{i=1}^n K_i u_i \quad (2-1-2)$$

式中 $K_i (i = 1, 2, 3, \dots, n)$ 为每一输入量的传递系数，这就实现了 n 个输入量按比例相加的运算。

三、积分求和器

积分求和器的其原理线路图如图 2-1-3 所示。积分求和器的输入电压与输出电压的关系为

$$u_0 = -\left(\frac{1}{R_1 C_f} \int u_1 dt + \frac{1}{R_2 C_f} \int u_2 dt + \frac{1}{R_3 C_f} \int u_3 dt + \cdots + \frac{1}{R_n C_f} \int u_n dt\right) = \sum_{i=1}^n K_i \int u_i dt$$

式中 $K_i (i = 1, 2, 3, \dots, n)$ 为积分求和器的传递系数，这就实现了 n 个输入量按积分求和的运算。

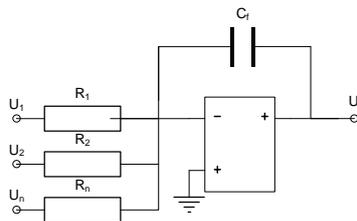


图 2-1-3 积分求和线路

所谓非线性特性，是指它的输出与输入信号之间的关系是非线性。在实际物理系统中，往往许多部件在不同程度上都具有非线性特性。常见的有饱和特性、继电器特性、死区特性。下面针对典型非线性部件的模拟方法及原理分别加以说明。

一、饱和特性

饱和特性的模拟线路图及输入、输出特性如图 2-2-1 所示。它是由比例

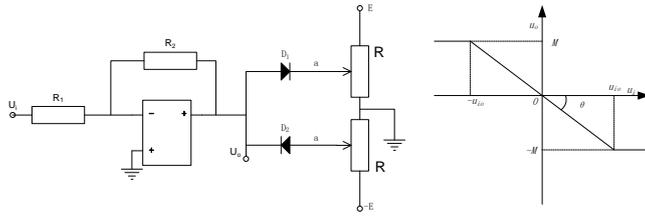


图 2-2-1 饱和特性线路及其输入输出特性

运算部件及输出限幅器两部分组成。当输入信号 u_i 在 $-u_{i0} < u_i < u_{i0}$ 范围内变化，其输出与输入成线性关系；当 $|u_i| > |u_{i0}|$ ，其输出保持一常量 M 或 $-M$ 。上述关系可用如下数学表达式表示：

$$u_o = \begin{cases} Ku_i & |u_i| < u_{i0} \\ -M \text{sign} u_i & |u_i| \geq u_{i0} \end{cases} \quad (2-2-1)$$

其中 $u_i > u_{i0}$ 时， $\text{sign} u_i = 1$ ； $u_i < -u_{i0}$ 时， $\text{sign} u_i = -1$

式 2-2-1 中的饱和值 M 等于整定的限幅值，增益 K 等于反馈电阻 R_f 与输入电阻 R_i 之比。在一般的情况下，系统中由于饱和特性的非线性部件存在，使系统的开环增益降低，从而增大系统的过渡过程时间和稳态误差，但在某些控制系统中，人们有目的地引入饱和非线性环节，例如在 SCR—D 双闭环直流调速系统中，其速度调节器和电流调节器均设计成具有饱和特性，以改善系统的动态性能和限止系统的最大电流。

二、继电器特性

继电器特性的模拟线路图及输入—输出特性如图 2-2-2 所示。由于运算放大

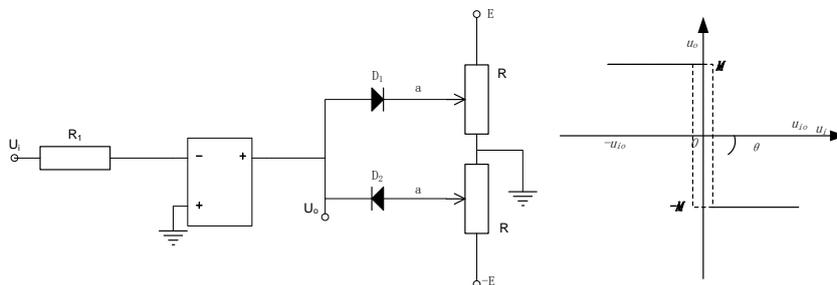


图 2-2-2 继电器特性线路图与输入输出特性图

器的反馈回路断开，其增益趋向于无限大，因此只要非常小的输入信号，其输出就立即增至限幅值 M 它可用如下数学表达式表示

$$u_o = -M \text{sign} u_i \quad |u_i| > 0 \quad (2-2-2)$$

实际上由于运算放大器不可能是理想的，即电子元件和电路有小惯性(尽管非常小)，而且通频带有限，因此实际的继电器特性是有小回环。

三、死区特性

死区特性的模拟线路图及输入—输出特性如图 2-2-3 所示，即在比例运算

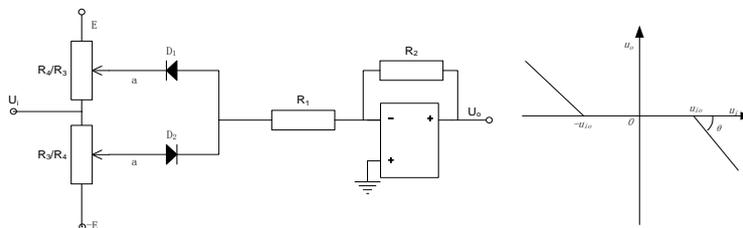


图 2-2-3 死区特性线路图与输入输出特性

放大器的输入回路中接入一个二极管钳位电路。当输入信号 u_i 较小时， D_1 、 D_2 均不导通，其输出电压 u_o 为零；当输入信号大于死区宽度 u_{io} 时，其输出与输入成线性关系。上述关系可用如下数学表达式示

$$u_o = \begin{cases} 0 & |u_i| \leq u_{io} \\ -K(u_i - u_{io} \operatorname{sign} u_i) & |u_i| > u_{io} \end{cases} \quad (2-2-3)$$

式中：

K ---线性段的增益；

$2u_{io}$ ---死区的宽度；

u_{io} ---二极管导通的临界电压

在自动控制系统中具有死区特性的部件较多。例如同步电机由于要克服摩擦和负载转矩，因此需要有一定的启动电压，这个电压称作同步电机的死区。显然，摩擦和负载转矩越大，同步电机的死区也越大；又如控制系统中的检测部件如测速发电机、自整角机和旋转变压器等，由于其电刷和换向器的接触压降也会产生死区。在死区范围内，系统处于开环运行状态，从而失去了自动调整的功能，将使系统的动态性能变差和稳态误差增大。因此必须提高部件运动部分的润滑度，改进元件的加工工艺及选用优质材料等措施，以减小其非线性死区。但有时利用死区特性抑制某种干扰，从而使系统提高抗干扰的能力。

§ 2—3 典型环节的模拟及参数测试

根据数学模型的相似原理，我们应用电子元件模拟工程系统中的典型环节，然后加入典型测试信号，测试环节的输出响应。反之从实测的输出响应也可以求得未知环节的传递函数及其各个参数。

模拟典型环节传递函数的方法有两种：第一种方法，利用模拟装置中的运算部件，采用逐项积分法，进行适当的组合，构成典型环节传递函数模拟结构图；第二种方法将运算放大器与不同的输入网络、反馈网络组合，构成传递函数模拟线路图，这种方法可以称为复合网络法。本节介绍第二种方法。

一、惯性环节的模拟

惯性环节的传递函数为

$$G(s) = \frac{C(s)}{R(s)} = \frac{K}{T_s + 1} \quad (2-3-1)$$

其中：

K—静态放大倍数

T—惯性时间常数

惯性环节的模拟电路如图 2-3-1 所示。模拟电路的传递函数为

$$G(s) = \frac{U_o(s)}{U_i(s)} = \frac{R_2}{R_1} \frac{1}{R_2 C s + 1} \quad (2-3-2)$$

比较 (2-3-1) 式和 (2-3-2) 式，得 $K=R_2/R_1$ ， $T=R_2C$

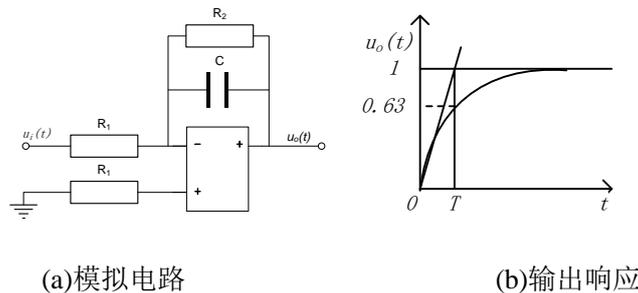


图 2-3-1 惯性环节的模拟电路及响应

当输入负阶跃信号时，其输出响应如图 2-3-1(b)所示。从图中可知，**T** 和 **K** 是响应曲线的两个特征量。**T** 表示输入阶跃信号后，响应按指数上升的快慢，它可从响应曲线实测得到。

二、积分环节的模拟

积分环节的传递函数为

$$G(s) = \frac{C(s)}{R(s)} = \frac{1}{T_i s} \quad (2-3-3)$$

其中 T_i —积分时间常数.

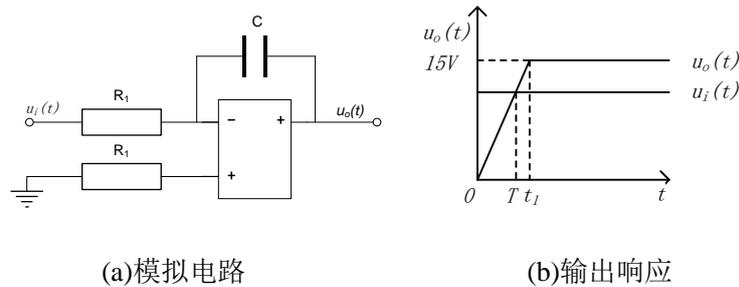


图 2-3-2 积分环节的模拟电路及响应

积分环节的模拟电路图如图 2-3-2(a)所示，模拟电路的传递函数为

$$G(s) = \frac{u_o(s)}{u_i(s)} = -\frac{1}{RCs} \quad (2-3-4)$$

比较 (2-3-3) 和 (2-3-4) 二式，得 $T_i = RC$

当输入负阶跃信号时，其输出响应如图 2-3-2(b)所示。从图中可知，积分时间常数 T_i 是积分环节的特征量，它表示阶跃输入后响应按线性上升的快慢， T_i 可从响应曲线上求出，即响应上升到阶跃输入幅值时所需的时间。积分环节的特点是，不管输入幅值多小，输出就不断地按线性增长，输入幅值愈小，增长的速率愈小，只有输入为零时，输出才停止增长而保持其原来的数值。从图中可看出运算放大器最终达到饱和值。

三、比例积分环节的模拟

比例积分环节的传递函数为

$$G(s) = \frac{C(s)}{R(s)} = K + \frac{1}{T_i s} \quad (2-3-5)$$

其中 K —比例系数； T_i —积分时间常数

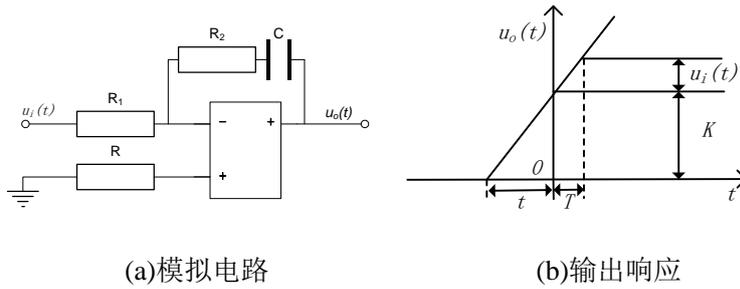


图 2-3-3 比例积分环节的模拟电路及响应

比例积分环节的模拟电路图如图 2-3-3(a)所示。模拟电路的传递函数为

$$G(s) = \frac{u_o(s)}{u_i(s)} = -\left(\frac{R_2}{R_1} + \frac{1}{R_1 C s}\right) \quad (2-3-6)$$

比较 (2-3-5) 和 (2-3-6) 两式得

$$K = \frac{R_2}{R_1} \quad T_i = R_1 C$$

当输入负阶跃信号时，其输出响应如图 2-3-3(b)所示。从该图中可以得到比例积分环节的特征参数 K 和 T_i 。必须注意：在测试积分环节和比例积分环节的阶跃响应时，由于存在储能元件 C ，因此每次输入阶跃响应时，必须保证 u_c 为零，否则将因 u_c 的初始值不同使每次测得的响应不同。

四、比例积分微分环节的模拟

比例积分微分环节的传递函数为

$$G(s) = \frac{C(s)}{R(s)} = K + \frac{1}{T_i s} + T_d s \quad (2-3-7)$$

K —比例系数； T_i —积分时间常数； T_d —微分时间常数。

该环节的模拟电路如图 2-3-4 所示，当满足 $R_1 \gg R_3$ ，

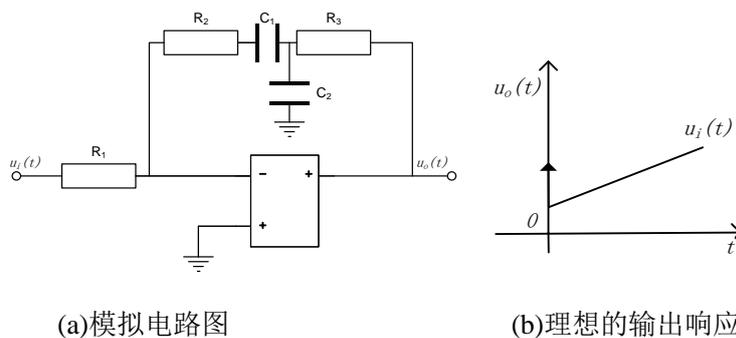


图 2-3-4 比例积分微分环节的模拟电路及理想的响应

$C_2 \gg C_1$ 时, 该电路的传递函数为

$$G(s) = \frac{U_o(s)}{U_i(s)} = - \left(\frac{R_2 C_1 + R_3 C_2}{R_1 C_1} + \frac{1}{R_1 C_1 s} + \frac{R_2 R_3 C_2}{R_1} s \right) \quad (2-3-8)$$

比较 (2-3-7) 和 (2-3-8) 两式得

$$K_p = \frac{R_2 C_1 + R_3 C_2}{R_1 C_1} \quad T_i = R_1 C_1 \quad T_d = \frac{R_2 R_3 C_2}{R_1}$$

对于理想的比例积分微分环节, 当输入负阶跃信号时其输出响应如图 2-3-4(b)所示, 在输入跃变时, 它的输出响应能够以无限大的变化率在瞬间跃至 ∞ , 又在此瞬间下降至按某一比例 K_p 分配的电压值, 并立即按积分时间常数 T_i 规律线性增长。而模拟比例积分微分环节的输出响应, 在输入跃变时只能以有限的变化率上升至运算放大的饱和值就不再增长, 经过一段时间, 又以有限的变化率下降。这是因为模拟电路是在满足 $R_1 \gg R_3$ 、 $C_2 \gg C_1$ 的条件下, 忽略了小时间常数才得到近似的 *PID* 数学模型式, 而且运算放大器也不是理想的, 因此实际比例积分微分环节的响应曲线与图 2-3-4(b)略有不同。

综上所述, 典型环节的模拟方法是: 根据典型环节的传递函数, 选择适当的网络作为运算放大器的输入阻抗与反馈阻抗, 使模拟电路的传递函数与被模拟环节的传递函数具有同一表达式, 然后根据被模拟环节传递函数的参数, 计算出模拟电路各元件的参数值。

§ 2—4 控制系统的模拟

在控制系统的研究中，除了用微分方程来描述系统的特性外，还常常采用各种典型环节组成的方框图形式来表示系统的特性，因此控制系统的模拟方法有两种：一种是根据系统的微分方程进行模拟；一种是根据系统的方框图进行模拟。

一、根据微分方程的模拟

微分方程模拟的基本原理就是采用逐项积分法。首先将方程中的最高阶微分项分离出来，输入到一串积分器中进行逐项积分，从而依次得到各低阶微分项。然后按照方程的各项系数将每个积分器的输出反馈到输入端进行代数相加，形成闭合回路，使得模拟电路图的方程与微分方程的关系一致。

下面以二阶系统为例，具体说明模拟方法。二阶系统又称振荡环节，其传递函数的标准形式为

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (2-4-1)$$

式中 ξ —— 阻尼比； ω_n —— 无阻尼自然振荡频率。

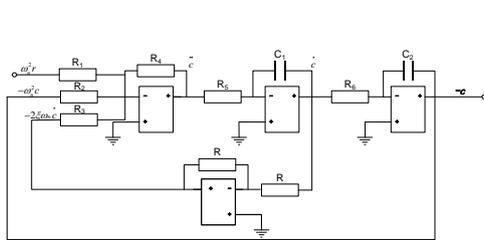


图 2-4-1 二阶系统的模拟电路图

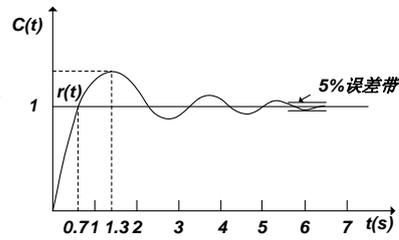


图 2-4-2 二阶系统的阶跃响应

将式 (2-4-1) 改写成微分方程为

$$\ddot{c} = -2\xi\omega_n \dot{c} - \omega_n^2 c + \omega_n^2 r \quad (2-4-2)$$

式(2-4-2)可用一只三输入的加法器来模拟，加法器的输出即是 \ddot{c} ，加法器的三个输入端就是式 (2-4-2) 右端的三个系数。它的模拟电路如图 2-4-1 所示。在模拟电路中，只要改变加法器的三个输入电阻，二阶系统的两个特征参数 ξ 和 ω_n 就随着改变，系统的输出响应也随之

变化。若在模拟电路的输入端加一负单位阶跃信号，实际测量的阶跃响应曲线如图 2-4-2 所示，则可直接得到系统的时域性能指标。

二、根据方框图的模拟

首先根据控制系统的方框图分解成若干典型环节，再按每个典型环节设计模拟电路，最后将各个典型环节按方框图连接起来，就得到控制系统的模拟电路图。

如一位置随动系统的方框图如图 2-4-3 所示。根据该方框图可以分成四个典型环节：一个信号综合环节、二个惯性环节和一个积分环节。然后按每个环节设计模拟电路并选择元件参数，即 $R_2/R_1=K_1$ ， $R_4/R_3=K_2$ ， $R_4C_1=T_1$ ， $R_3C_2=T_2$ ， $R_6C_3=1$ 。该系统的模拟电路如图 2-4-4 所示，图中， $\Delta \theta$ 模拟系统的角差， n 模拟系统的转速。可见，当已知系统的数学模型时，就可画出系统的模拟方框图。但值得注意的是：(1)已知系统的方框图，可分解成不同的典型环节，使用运算放大器的数目就不同。即系统模拟电路图不是唯一的。(2)每经过一个运算放大器，符号要改变一次。因此，如果回路中的运算放大器数目是偶数时，那末就会形成正反馈，变成正反馈系统；输出电压将要一直增加到饱和值为止，总之在模拟电路图中，每一个回路的运算放大器数目必须是奇数。

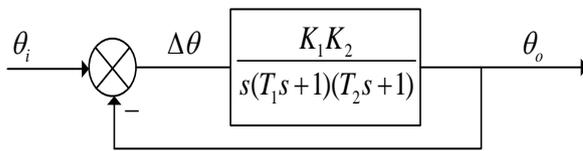


图 2-4-3 系统方框图

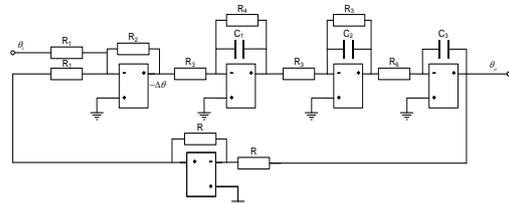


图 2-4-4 系统模拟电路图之一

随动系统

§ 3—1 概述

随动系统又称伺服系统或伺服机构，它要解决的主要问题是实现执行机构对位置指令(给定量)的准确跟踪。在随动系统中，输出量一般是机械量，例如机械位移、速度或加速度等等，反馈装置将输出量变换成与输入量相同的信号，然后进行比较，得出偏差，系统是按照偏差的性质(大小及符号)进行控制的，控制的结果是减少或消除偏差，使系统的输出量准确地跟踪或复现输入量的变化。因此随动系统是一种反馈控制系统，系统中的位置指令(给定量)和被控制量一样都是位移(或代表位移的电量)，可以是角位移，也可以是直线位移。

随着控制技术的发展，随动系统已经广泛地应用于军事工业和民用工业。例如机械制造工业中仿型机床的跟踪控制和数控机床的加工轨迹控制；造纸工业中纸张与卷筒之间的同步协调转动；仪器仪表工业中的电位差计、函数记录仪等等。虽然随动系统的种类繁多，但其闭环结构的基本特征是位置反馈。根据位置给定信号和位置反馈信号以及两个信号的综合比较来分类，可以分成模拟式随动系统和数字式随动系统。数字式随动系统根据不同的位置信号检测元件，又可分成数字式相位控制随动系统、数字式脉冲控制随动系统和数字式编码控制随动系统。本章简单介绍模拟式随动系统的结构原理、典型部件和控制方法。

§ 3—2 随动系统的结构原理

位置随动系统是一种位置反馈控制系统，因此一定具有位置给定和位置反馈的检测装置。通过位置给定装置将希望的位移转换成具有一定精度的电量，利用位置反馈装置随时检测出被控机械的实际位移，也把它转换成具有一定精度的电量，与给定进行比较，把比较得来的偏差信号放大以后，控制执行电机向消除偏差的方向旋转，直到达到一定的精度为止。这样，被控制机械的实际位置就能跟随给定变化，构成一个位置随动系统。

实验随动系统采用的位置给定和位置反馈检测装置是一对旋转变压器，其检测精度比较高，电气误差角一般在 $5\sim 20$ 角分；采用的执行电机是直流力矩电机，由于该电机能长期低速运行，故拖动负载时不需减速器。其系统原理图如图 4-2-1 所示。它用一对正余弦旋转变压器作为位置输入和位置反馈的检测装置，也就是说通过位置输入检测装置将希望移动的位移量转换成具有一定精度的电量，利用位置反馈检测装置随时检测执行电机的实际位移，并

将它转换成一定精度的电量，此电量反馈到系统的输入端与输入电压进行比较，比较后得到的偏差经相敏整流器，再经校正装置和功率放大器放大，从而推动执行电机向消除偏差的方向旋转，因此执行轴的位置总是跟踪输入轴的位置的变化而变化。

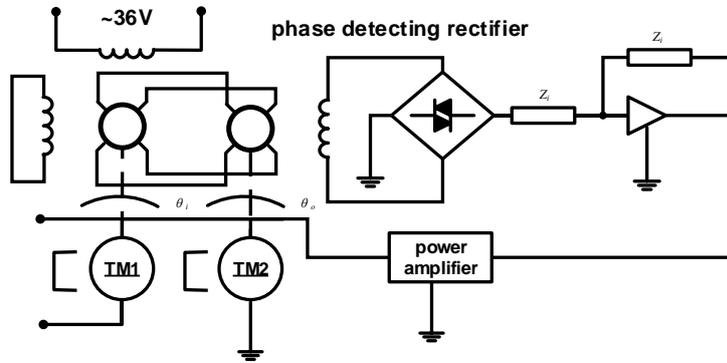


图 3-2-1 采用旋转变压器的位置随动系统原理图

§ 3—3 随动系统的主要部件及其传递函数

随动系统的结构形式多种多样，但从结构功能上看，都有很强的共性，事实上，任何一个随动系统，都有角度(或速度)测量元件、信号变换元件、功率放大元件以及执行元件等几个组成部分，如图 3-3-1 所示。

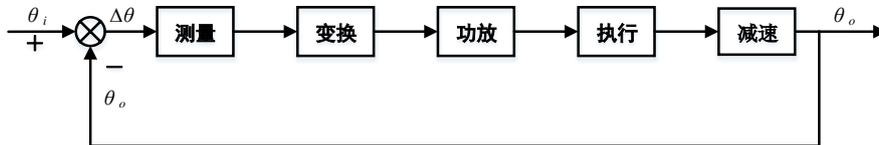


图 3-3-1 随动系统功能方框图

其中 θ_i —输入轴的转角； θ_o —输出轴的转角； $\Delta\theta = \theta_i - \theta_o$ —系统的角差

随着机电及电子器件的不断发展与完善，上图中的每个环节均采用多种不同的元器件来实现。例如，采用电位器、自整角机、旋转变压器、或光电编码盘作为角度测量装置，采用交磁放大机、晶体管放大器、脉冲调宽放大电路作为功率放大环节，采用交流测速发电机、测速电桥作为角速度测量装置，变换元件通常指的是调制器、解调器、相敏整流器。

实验随动系统的线路图如图 3-3-2 所示。对于一个实际系统如何确定它的数学模型呢？一般需经过下列步骤：

1. 分析系统的工作原理和每个部件的功能，并确定系统的输入量和输出量后，画出系统的功能方框图，如图 3-3-3 所示。

2. 对系统及每个部件确定出既不影响运动过程本质特征，又能使数学模型简化的条件，并确定每个部件的输入量和输出量，然后根据机理或实验数据确定每个部件的数学模型。
3. 将每个部件的数学模型填入功能方框图中的相应方框中，得到系统方框图，

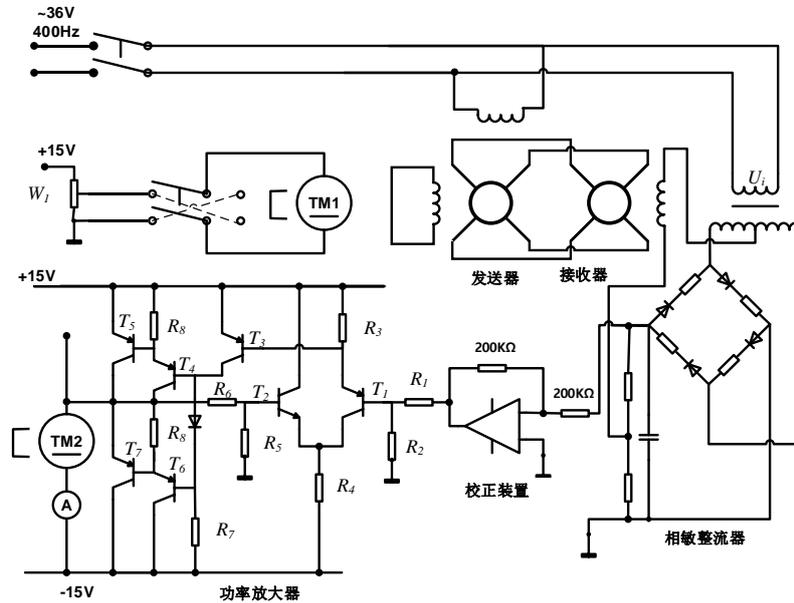


图 3-3-2 实验随动系统的线路图

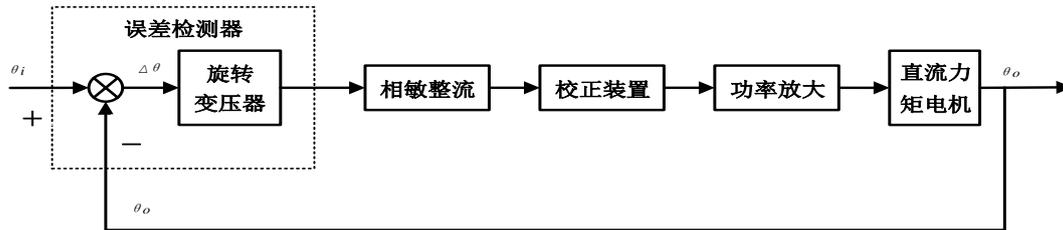


图 3-3-3 实验随动系统的功能方框图

下面根据上述步骤先简单介绍该系统中各主要部件的工作原理。

一、正余弦旋转变压器

正余弦旋转变压器是一种高精度的控制电机，用来测量输入轴与执行轴转角的传感器。它的原理和普通变压器一样，但结构与普通变压器不同，而与一般的控制电机一样，由定子和转子组成，定子由两个匝数相同，空间位置互相垂直的绕组组成，转子也由两个匝数相同，空间位置互相垂直的绕组组成，定子和转子间的气隙磁场为严格的正弦分布。定子的两个绕组有四个输出端 D_1 、 D_2 、 D_3 、 D_4 ；转子的两个绕组也有四个输出端 Z_1 、 Z_2 、 Z_3 、 Z_4 ，它们分别与四个滑环连接，由四个电刷引出电信号。在随动系统中一般用一对旋转变压器（BR）来检测输入轴与执行轴之间的误差，它们的测角原理线路如图 3-3-4 所示，图中

转子绕组对应整步连接，与输入轴相联的称为发送器(BRT)，它产生并发送角度位置信号。与输出轴相联的称为接收器(BRR)，它反映跟踪电机的实际位置信号。假设两旋转变压器的参数完全相同，若对发送器的定子绕组施加激磁电压为

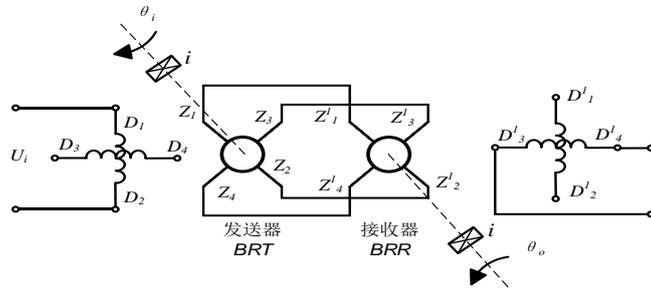


图 3-3-4 旋转变压器测角线路

$$u_i = U_m \sin \omega t$$

如果忽略旋转变压器所产生的微小差异，则接收器转子的输出误差电压为一载波调制信号，表达式为：

$$u_o = U_m \sin(\theta_i - \theta_o) \sin \omega t \quad (3-3-1)$$

式中： $\omega = 2\pi f$ ， f 称载波频率，一般常用 400Hz 到 500Hz。

从式(3-3-1)可以看出：角差线路的输出电压 u_o 的相位与激磁电压 u_i 成正比， u_o 的幅值与正弦成正比。 u_o 不仅反应角差 $\Delta \theta$ ，而且能反应角差的极性。

在实验随动系统中，若系统的动态误差角和静态误差角均小于 20° ，则检测装置的误差电压 u_o 与误差角 $\Delta \theta$ 近似为线性关系，即

$$u_i = U_m \Delta \theta \sin \omega t \quad (3-3-2)$$

从上式可知，在满足 $\Delta \theta < 20^\circ$ 的条件下，旋转变压器的测角线路，可近似为一比例环节。公式(3-3-2)可以通过实验来验证。若使发送器匀速旋转，接收器静止，则角差 $\Delta \theta$ 将从 0° 到 360° 之间周期性地均匀变化。我们用示波器观察接收器的



图 3-3-5 $u_o = f(\Delta \theta)$ 波形

输出电压 u_o 的波形，如图 3-3-5 所示，其中交流载波频率与发送器激磁电压频率相同，其幅值与 $\Delta \theta$ 的正弦成正比。交流载波包络线的频率与发送器的转速有关，转速愈高，包络线的频率愈高。从上所述，用一对正余弦旋转变压器检测误差角的输出电压原理相当于乘积调制器原理，即调制器的输信号是 u_o 是输入信号 $U_m \sin(\Delta \theta)$ 与交流载波信号 $\sin \omega t$ 的乘积。

二、相敏整流器

实验随动系统采用二极管全波相敏整流器，它实质上是一种开关解调器(又称同步解调器)，其原理线路如图 3-3-6 所示。其中四个参数相同的二极管 D_1 、 D_2 、 D_3 、 D_4 与四个阻值相同的电阻 R_1 、 R_2 、 R_3 、 R_4 按同一方向串联成一闭合环路，环路的四个桥臂是对称的。在它的的一个对角线上加解调激励电压 u_t ，在另一个对角线上加输入信号电压 u_i (即随动系统测角线路的接收器定子输出的误差电压)。一般要求激励电压 u_t 的幅值要比输入信号 u_i 的幅值大一倍以上。桥臂中的电阻为限流电阻。

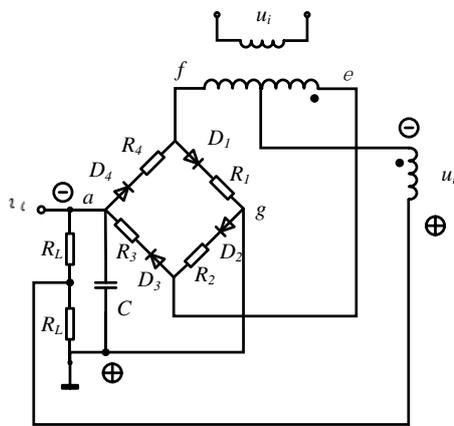


图 3-3-6 相敏整流器原理图

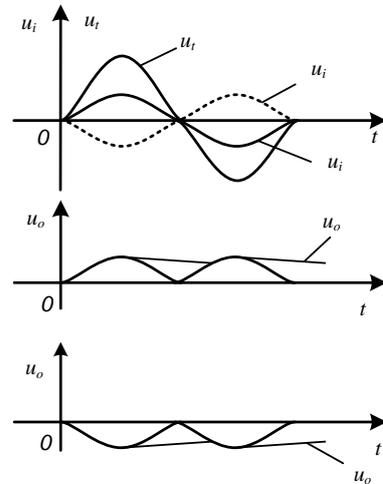


图 3-3-7 相敏整流电压波形

激励电压 u_t 起着开关作用，使二极管导通和截止。当输入交流信号 $u_i=0$ 时，在激励电压的正半周时(如图 3-3-6 中的 e 为正， f 为负)，二极管 D_3 、 D_4 导通， D_1 、 D_2 截止。由于环路的四个桥臂和变压器次级绕组都是对称的，所以当 D_3 、 D_4 导通时， a 点 g 点是等电位，同理，在激励电压的负半周时(e 为负 f 为正)， a 点和 g 点也是等电位，所以当输入信号 u_i 为零时，尽管开关不断地动作，二极管中有电流流过，但负载电阻 R_L 中无电流流过，输出电压仍然为零。

当输入交流信号 $u_i \neq 0$ ，假定它与解调激励信号 u_f 是同相位(如图 3-3-7)所示的同名端)，在电源正半周时，在 u_i 的作用下，电桥的右两臂 D_1R_1 、 D_2R_2 截止。左两臂 D_3R_3 、 D_4R_4 导通。输入信号 u_i 极性上正、下负，电流经 u_f 的右半边绕组、 D_3R_3 及负载 R_L 形成闭路。输出电压 $u_o = I_L R_L$ 的极性上正、下负，或 u_o 为正，同理在电源负半周时，电桥右两臂 D_1R_1 、 D_2R_2 导通，左两臂 D_3R_3 、 D_4R_4 截止，在输入信号 u_i 作用下，电流经 R_L 、 D_2R_2 及 u_i 的右半边绕组形成闭路，输出电压 $u_o = I_L R_L$ 的极性仍然是上正下负，即 u_o 正。

如果输入交流信号 u_i 的相位与 u_f 的相位差 180° (在图 3-3-6 中以+、-标记)，则在电源正半周时，仍然是电桥左两臂导通，右两臂截止，在 u_i 作用下，电流经 R_L 、 D_4R_4 及 u_i 的左半边绕组构成闭合回路，电流 I_L 改变了方向，自下而上流过 R_L ，所以输出电压 u_o 改变了极性，或称 u_o 为负。同理，在电源负半周时， u_o 极性也是为负。

旋转变压器测角线路的输出电压波形和相敏整流器的输出电压波形可通过实验来验证，将实验随动系统开环(跟踪电机 TM2 不接)，并驱动电机 TM1，即误差角在 0° 与 360° 之间变化。

此时将接收器输出电压接至相敏整流器的输入端，并将其输入电压和输出电压分别接到示波器，得到的波形图如图 3-3-8 所示。 u_o 中有 800Hz，说明电容滤波不够理想。

从图 3-3-8 中的输入与输出电压波形图，还可求出二者的相移角为

$$\varphi = -360^\circ \frac{b}{a} \quad (3-3-3)$$

说明相敏整流器的输出电压迟后其输入电压的相位角为 φ 。

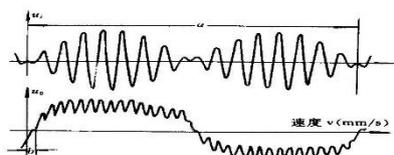


图 3-3-8 相敏整流器的输入输出电压波形

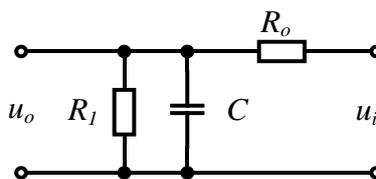


图 3-3-9 相敏整流器的等效电路图

由于相敏整流器中四只二极管是非线性元件，其正反向电阻随其工作点的变化而变化，在理论推导该环节的传递函数时，用如图 4-3-9 所示的等效电路来分析，从输出端来看，有一个内阻 R_o ($R_o \neq$ 常数)，该电路的传递函数为

$$G_2(s) = \frac{U_o(s)}{U_i(s)} = \frac{K_2}{T_c s + 1} \quad \text{式中} \quad K_2 = \frac{R_L}{R_L + R_o} \quad T_c = \frac{R_L R_o}{R_L + R_o} C$$

由于滤波时间常数 T_c 很小，故可近似：

$$G_2(s) = K_2 \quad (3-3-4)$$

三、校正装置

在实际反馈控制系统中，被控对象的络构和参数往往已经确定如果仅仅以误差信号 $e(t)$ 来调节系统，一般来说是难以满足用户的技术要求。因此，需要对原系统加入合适的校正装置，校正装置可分为并联校正、串联校正及复合校正三类，采用哪一类校正方式，取决于系统中被控对象本身的物理结构和动态特性，对系统性能的要求以及可采用的元件等实际情况。一般来说，串联校正比较简单，调试也较容易，若采用有源串联校正，还能起到前后级的隔离作用。

校正装置在控制系统中联接位置的选择原则上应考虑两点，第一在传递功率较小的电路中，即在系统主通道的前半部，不能串接在系统的功率放大部分。第二，直流校正网路只能接在传递直流信号的电路中。本实验随动系统的串联校正装置的联接位置介于相敏整流器与功率放大器之间。

串联校正采用得比较普遍。其作用是将系统中的误差信号 $e(t)$ 加工成合适的控制信号。校正装置所进行的运算叫做控制规律，并且根据各种校正装置所实现的具体控制规律把它们分别称为比例控制器、比例积分控制器和比例积分微分控制器等。

下面分别说明几种基本控制器的作用。

1. 比例 (P) 控制器

比例控制器的传递函数为

$$G(s) = K_p \quad (3-3-5)$$

比例控制器的输出信号 $u(t)$ 成比例地、无延迟地反应其误差信号 $e(t)$ 即

$$u(t) = K_p e(t) \quad (3-3-6)$$

它的作用是增大系统的开环比例系数，以提高系统的稳态精度，降低系统的惰性，加快响应速度。但过大的 K_P 通常会造成系统的动态性能变坏，甚至不稳定，因此比例控制器通常不单独使用。

2. 积分(I)控制器

积分控制器的传递函数为

$$G(s) = \frac{1}{T_i s} \quad (3-3-7)$$

它的输出量 $u(t)$ 是输入量 $e(t)$ 对时间的积分，即

$$u(t) = \frac{1}{T_i} \int_0^t e(t) dt \quad (3-3-8)$$

它的特点是：当输入信号 $e(t)$ 变为零以后，控制信号 $u(t)$ 如不为零，即可以维持在某一非零值。系统中加入积分控制器后，可以提高系统的型别，以达到消除系统稳态误差的目的。但积分校正却使系统的动态性能变坏，甚至会导致系统的不稳定，此外积分控制作用往往导致系统响应迟缓。

3. 比例微分(PD)控制器

比例微分控制器的传递函数为

$$G(s) = K_p(1 + \tau s) \quad (3-3-9)$$

PD 控制器的输出信号 $u(t)$ 既成比例地反应输入信号 $e(t)$ ，又成比例地反映输入信号的导数，即

$$u(t) = K_p e(t) + K_p \frac{de(t)}{dt} \quad (3-3-10)$$

其中 K_p — 比例系数 τ — 微分时间常数

微分控制作用有时也叫做速率控制，即输入信号的变化越大，输出信号越大。微分时间常数，是速率控制作用超前于比例控制作用的时间间隔，也就是说微分控制作用具有预测的特性。但是当动态过程接近于稳态时，误差信号变化不大或是变化缓慢，微分作用也就微不足道，所以微分作用不能单独使用，它总是与比例作用结合起来使用。

4. 比例积分(PD)控制器

比例积分控制器的传递函数为

$$G(s) = K_p \left(1 + \frac{1}{T_i s}\right) = \frac{K_p(1 + T_d s)}{T_i s} \quad (3-3-11)$$

其输出信号 $u(t)$ 同时成比例地反应输入信号 $e(t)$ 及其积分

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt \quad (3-3-12)$$

式中 K_p —比例系数 T_i —积分时间常数

由于 PI 控制器有 K_p 和 T_i 两个可调参数, 适当加以选择就有可能使系统既稳定又具有较好的稳态和动态性能, 即兼有比例和积分两种控制器的优点。其中, 积分控制作用是提高系统的型别, 以消除或减小稳态误差, 而比例微分的作用是提高系统的阻尼程度, 从而保证闭环系统的稳定性。

5. 比例积分微分(PID)控制器

PID 控制器的传递函数为

$$G(s) = K_p \left(1 + \frac{1}{T_i s} + \tau s \right) \quad (3-3-13)$$

它是一种由比例、积分、微分等基本控制规律组合而成的复合控制规律, 兼有几种单独控制器各自的优点。其运动方程式为

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p \tau \frac{de(t)}{dt} \quad (3-3-14)$$

PID 控制器的传递函数可改写为

$$G(s) = \frac{K_p}{T_i} \frac{T_i \tau s^2 + T_i s + 1}{s} \quad (3-3-15)$$

当 $4\tau/T_i < 1$ 时, 上式还可写成

$$G(s) = \frac{K_p}{T_i} \frac{(\tau_1 s + 1)(\tau_2 s + 1)}{s} \quad (3-3-16)$$

从上式看出, 比例积分微分控制规律既可使系统的型别提高, 又将提供两个负实零点, 它比 PI 控制器多提供一个负实零点, 从而进一步提高了系统的动态性能指标。因此比例积分微分控制规律在控制系统中得到了广泛应用。

四、功率放大器。

功率放大器的原理线路如图所示，其主要作用是给出足够的电流以满足驱动直流力矩电机的需要，同时还要具有较低的输出阻抗和较高的输入阻抗以起到放大器与力矩电机间的隔离作用。本实验采用 OCL 准互补推挽功率放大电路。

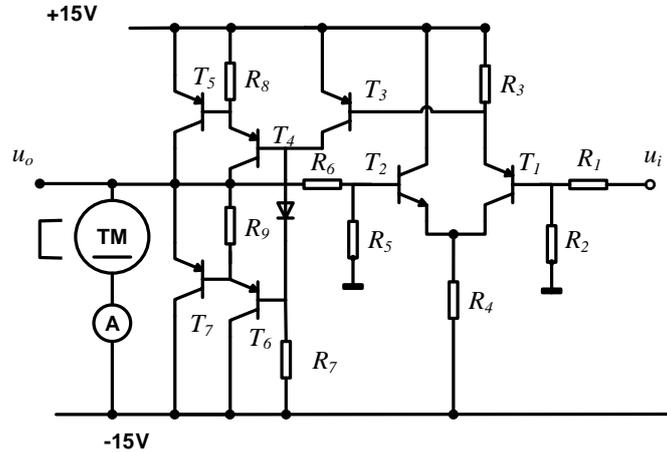


图 3-3-10 功率放大器电路图

功率放大器的电压放大倍数的计算公式为

$$K_4 = \frac{R_5 + R_6}{R_5} \quad (3-3-17)$$

五、直流力矩电机

一般的直流伺服电动机应用于控制系统时，因电动机转速高，往往要用齿轮减速后拖动负载，这样由于齿轮传动引起的齿隙误差，降低于系统的精度，为取消减速齿轮以改善其性能，可用长期低速运行的力矩电动机。它的工作原理与普通直流伺服电动机相同，为了在同样体积与电枢电压下降低转速、增加力矩，通常把电机作成扁平型，并做成多极永磁式，无须激磁。为了减小转矩脉动，电枢的槽数、换向片数和串联导体数设计得都比较多。这种电机低速转时可达每分钟几转，转速波动可在 5% 以内，并可长期在堵转状态下运行，输出足够的转矩而不致损坏电机。此外，机械特性、调节特性以及快速响应特性都比较好(机电时间常数一般只有十几毫秒至几十毫秒)。因此力矩电机适合于在位置伺服系统中作执行元件用。

力矩电机的电气原理如图 3-3-11 所示，磁场是永磁式的，其数学推导如下。由图 4-3-12 可知，电枢电路的微分方程为

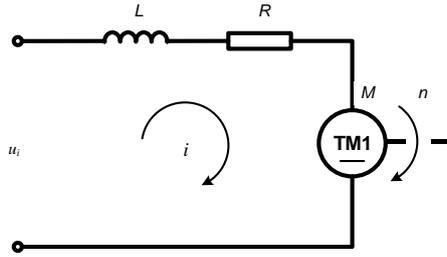


图 3-3-11 力矩电机的电器原理

$$u_i = L \frac{di}{dt} + Ri + C_e n \quad (3-3-18)$$

式中：L—电枢绕组的电感，H

i—电枢绕组中的电流，A

R—电枢绕组的电阻，Ω

C_e —电机的反电势常数，V/(r·min⁻¹)

n——电机的转速 r/min。

如果忽略电机轴上摩擦负载，则下式成立

$$M = C_m i = J \frac{dn}{dt} \quad (3-3-19)$$

式中：M—电机产生的转矩，kg·m

C_m —电机的力矩常数，kg·m/A；

J—电机和负载折合到电动机轴上的转动惯量 kg·m·s²

由(3-3-19) (3-3-20)式可得

$$\frac{u_i}{C_e} = T_M T_L \frac{d^2 n}{dt^2} + T_M \frac{dn}{dt} + n \quad (3-3-20)$$

式中： $T_m = \frac{JR}{C_e C_m}$ ——电机的机电时间常数

$T_L = \frac{L}{R}$ ——电机的电磁时间常数

若取电机的转速 n 为输出量，则由式(3-3-20)可求得力矩电机的传递函数为

$$G(s) = \frac{n(s)}{u_i(s)} = \frac{K_s}{T_M T_L s^2 + T_M s + 1} \quad (3-3-21)$$

式中： K_s —电机的传递系数，其值为 1 / C_e ， (r/min)·V。

对于小功率力矩电机，一般满足 $T_L \ll T_M$ ，因此上式可近似为

$$G(s) = \frac{n(s)}{u_i(s)} = \frac{K_5}{T_M s + 1} \quad (3-3-22)$$

若以角度 θ_0 作为输出量，则直流力矩电机的传递函数为

$$G(s) = \frac{\theta(s)}{u_i(s)} = \frac{K_5}{s(T_M s + 1)} \quad (3-3-23)$$

综上所述，随动系统中各部件的数学模型比较简单。但是，这种简单的数学模型是高度理想化后的抽象结果。它只有满足一定条件时，才能描述实际物理过程的本质特征。所以在这个数学模型的基础上应用控制理论，也必然应当遵守这些理想化的约定条件，在分析随动系统的性能指标时，应满足如下理想化的假设条件：在一定工作范围内(误差角在 200° 以内)，忽略旋转变压器和力矩电机的死区非线性；也不考虑运算放大器及功率放大器的饱和非线性。当系统

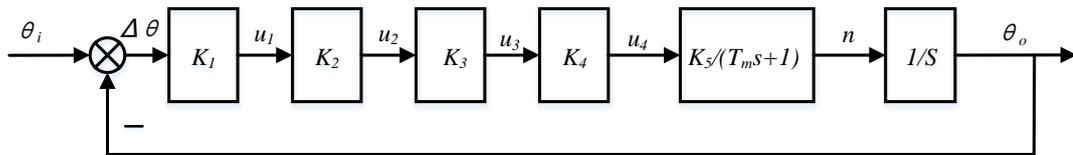


图 3-3-12 实验随动系统的方框图

开环放大系数比较小时，忽略小惯性时间常数的次要极点影响，也就是忽略了力矩电机电磁时间常数 T_L 及相敏整流器输出的滤波时间 T_c ，这样系统的方框图如图 3-2-12 所示。系统的数学模型为

$$G(s) = K_1 K_2 K_3 K_4 \frac{K_5}{s(T_M s + 1)} = \frac{K}{s(T_M s + 1)} \quad (3-3-24)$$

式中：K—系统的开环放大系数， s^{-1}

K_1 —接收器输出电压与系统误差角之间的传递系数，V / 度，

K_2 —相敏整流器输出电压与输入电压的传递系数；

K_3 —运算放大器输出电压与输入电压的传递系数；

K_4 —功率放大器输出电压与输入电压的传递系数；

K_5 —力矩电机的输出转速与输入电压的传递函数，即反电势常数 C_e 的倒数 (r/min) · V 或 (度/s) · V。

从式(3-3-24)可知，实验随动系统是二阶系统。根据理论分析，只有当开环放大系数趋向于无穷大时，系统才可能出现等幅振荡，而实验研究时当 K 增大到某一值时，系统就出现等幅振荡，这说明近似的数学模型不够准确，使得数学模型的分析结果与实际物理系统的实验研究不能很好吻合。这是因为系统开环增益比较大时，不能忽略小惯性时间常数的次要极点的影响。

若考虑相敏整流器的滤波时间常数 T_c 和电机的电磁时间常数 T_L ，并采用工程上对控制系统中的小惯性时间常数的处理方法。设系统中有 N 个小惯性时间常数 T_1 、 T_2 、 \dots 、 T_N 。则可用一个惯性环节来等效处理，即

$$\frac{1}{Ts+1} = \frac{1}{(T_1+T_2+\dots+T_N)s+1}$$

因此实验随动系统的开环传递函数为

$$G(s) = \frac{K}{s(T_M s+1)(T_e s+1)} \quad (3-3-25)$$

其中： $T = T_L + T_C$

T_L ——力矩电机的电磁时间常数；

T_C ——相敏整流器的滤波时间常数；从式(3-2-25)可知，实验随动系统比较精确的数学模型是一个三阶系统，其稳定性可用图 4-3-13 的根轨迹图来分析，这与实验结果能够很好的吻合。理论上当系统的 K 增大到图 4-3-13 中的 A 点时，会出现等幅振荡。实际上，由于系统参数的微小变化，不可能长期稳定工作在 A 点。

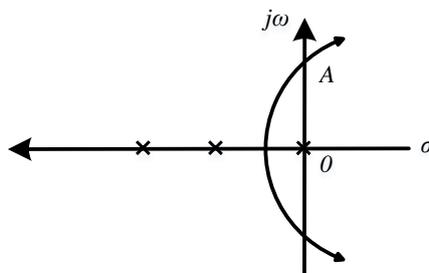


图 4-3-13 随动系统根轨迹图

建立一个物理系统后；必须进行系统的调试。只有通过调试，才能最后决定系统实际达到的技术指标。

系统调试所要解决的问题是：

1. 测试系统的各项性能指标，使其符合设计要求；
2. 获取系统的详尽数据，为今后改进该系统提供依据。

系统调试的一般步骤为：先部件后系统，先开环后闭环。即首先调整好每个部件，只有各部件处于正常工作状态时，方可对系统进行调试。系统调试的大致步骤为：反馈极性的判别，系统输入、输出指示的协调，调整校正网络的元件参数，使系统达到设计要求的性能指标。在系统的调试过程中，可能出现各种故障，我们需要灵活应用理论知识来分析故障原因，并及时排除故障，使系统进入正常运行状态，然后测量系统动态和稳态性能指标，下面结合实验随动系统简单介绍系统调试的过程。

一、部件及系统的零位调整

为了使随动系统输出的零位与输入的零位相一致，必须保证误差检测器的发送器位置为零时，接收器位置也为零。系统零位调整的步骤如下：按图 3-3-2 接好线后，将电位器 W1 逆时针方向旋到底，即速度输入为零，然后合上正、负 15 伏电源和 36 伏交流电源，形成闭环系统。轻微转动输入轴，使跟踪电机的负载电流为零，此时系统的输入指针与输出指针应指向同一刻度，否则，用小螺丝刀调整某一指针，使两指针位置协调一致。如果两指针相差 180 度，则可采取对调接收器输出绕组的极性或相敏整流器中同步变压器原边绕组的极性，也可对调跟踪电机的电枢绕组的极性。

二、部件静特性的测量方法

测量部件静特性的方法为：对部件输入一输入量，测量其对应的一个输出量，逐点测量一组数据。这种方法称为逐点测量法，例如测量力矩电机的传递系数，输入量是电枢的电压，输出量是电机的转速。首先在力矩电机的电枢两端接入一电压可调的信号源 u_i 和用来测量输入电压大小的电压表，并用秒表测量输出转速的大小，给定一个输入电压 u_i ，测量一个输出转速 n 然后画出一条 u_1 与 n 之间的曲线，再取其斜率就是力矩电机的传递系数，计算时应注意量纲。

测量数据的准确程度与测试方法和使用的测量仪表有直接关系。测试随动系统的误差检测器、相敏整流器、运算放大器和功率放大器的传递系数时，采用同时测量的方法，具体的操作步骤是：首先使系统在功率放大器与跟踪电机之间断开，然后给定一个系统的位置输入信号 θ_i ，测量上述四个部件的输出电压，得出一组数据，最后求出四个传递系数 K_1 、 K_2 、 K_3 、 K_4 。这种测量方法有两个优点：一是考虑了前后级间的负载效应，使测得的传递系数接近于系统运行时的实际情况；二是由于前一级的输出量就是后一级的输入量，因此测量数据减少一半，可以避免读数误差。

三、动态性能指标的测量

随动系统的位置指令是经常变化的，它是一个随机变量，因此输出响应的快速性、灵活性、准确性是随动系统的主要技术指标。也就是说，当给定量随机变化时，系统能使被控制量准确地跟踪并复现给定量。这样，随动系统的分析与设计往往是根据一些典型输入信号下的响应来进行考虑。比较常见的随动系统输入信号有

位置输入（即阶跃输入）：位置输入信号的表达式为：

$$\theta_i(t) = \theta_m \times u(t) \text{ 其特征参数是信号的幅值 } \theta_m$$

速度输入（即斜坡输入）：速度输入信号的表达式为：

$$\theta_i(t) = At \text{ 其特征参数是信号变化的速度 } A$$

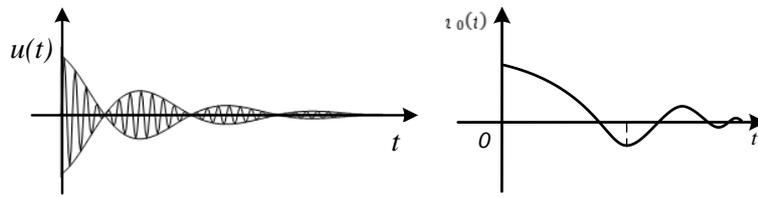
加速度输入（即抛物线输入）：加速度输入信号的表达式为

$$\theta_i(t) = Bt^2 \text{ 其特征参数是信号变化的加速度系数 } B$$

实验随动系统只测量阶跃响应和斜坡曲线。在阶跃输入作用下，随动系统的输出量是机械角，若用肉眼去观察接收器的指针是不易看清系统的各项动态性能指标的。因此必须将机械角变换成电信号后，才能测量系统的瞬态响应曲线，从§3—3节分析可知，误差检测器的功能，一是将系统的输入信号与输出信号进行比较，二是将系统的误差角转变成电信号。故可以通过测量接收器的输出电压 u_1 的波形，以间接方式来观察或记录系统的

瞬态响应曲线。因为系统的误差角 $\Delta\theta(t) = \theta_i(t) - \theta_0(t)$ ，当系统加入阶跃信号时， $\theta_i(t)$ 为一常数，因此 $\Delta\theta(t)$ 与 $\theta_0(t)$ 的变化规律相同，但由于 u_1 为载波调制信号如图3-4-1(a)所

示，图中除了误差调制信号外，还有 400Hz 的载波信号。一般取经过相敏整流器后的直流信号，如图 3-4-1 (b) 所示来测量阶跃响应。图 3-4-1 (b) 为随动系统的阶跃响应曲线。



(a) 相敏整流器的输入 (b) 相敏整流器的输出

3-4-1 随动系统的阶跃响应曲线。

四、稳态误差的测量

控制理论中分析的系统稳态误差称为原理性稳态误差，其大小与系统类型、输入函数的关系如表 3-4-1 所示，该误差没有包括执行电机的死区、测量元件的不灵敏区、运算放大器和功率放大器的零点漂移所产生的静差。通常系统的稳态误差比静差大得多；对于精度要求不高的随动系统，计算系统总的稳态误差时可忽略静差。但在高精度随动系统中，仅静差就满足不了系统的跟踪精度要求，必须增大系统的静态速度误差系数 K_P 以减小静差，我们用实验方法测量得到的稳态误差，包括了静差，因此往往大于理论计算的稳态误差。

表 3-4-1

输入信号 系统类型	单位阶跃 $r(t) = 1$	单位斜坡 $r(t) = t$
I 型	$ess = 0$	$ess = 1/K_e$
II 型	$ess = 0$	$ess = 0$

实际随动系统的输入信号，一般不是单位斜坡信号，此时系统的稳态误差不仅与静态速度误差系数 K_P 有关，而且与系统的跟踪速度有关，速度愈高稳态误差愈大。也就是说跟踪速度与稳态误差是一一对应关系。我们用实验方法测量系统的稳态误差，是指某一速度下的稳态误差，因此实测系统稳态误差时，必须记下对应的指令速度。实测稳态误差的方法是使随动系统等速跟踪运行，用秒表测得转速 n ，同时测量旋转变压器接收器的输出交流电压 u_1 ，然后就可

以根据旋转变压器测角线路的传递系数，求出系统稳态误差 e_{ss} 的值。例如实测的转速 n 为 $100(r/min)$, u_1 为 $0.5V$, 设旋转变压器测角线路的传递系数为 K_1 为 $1(V/度)$, 则该系统在 $100(r/min)$ 时, 测量的稳态误差为

$$e_{ss} = \frac{u_1}{K_1} = \frac{0.5}{1} = 0.5^\circ$$

理论计算某速度时的稳态误差为

$$e_{ss} = \frac{\Omega}{K_p}$$

式中 Ω —角速度; K_p —稳态速度误差系数

实验内容

实验一 典型环节的模拟实验

一、实验目的

1. 熟悉示波器的性能和使用方法；
2. 掌握典型环节的电模拟方法及其参数测试方法；
3. 测量典型环节的阶跃响应曲线，了解参数变化对动态特性的影响。

二、实验仪器

1. 控制理论实验箱一台；
2. 示波器一台；
3. 万用表一只。

三、实验原理

以运算放大器为核心元件，与 $R - C$ 输入网络和反馈网络组成各种典型环节，如图 1-1 所示。图中 Z_1 和 Z_2 为复数阻抗，它们都是由 R 、 C 构成。

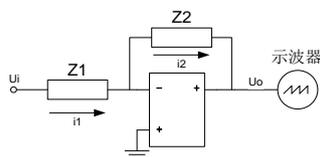


图 1-1 运放的反馈连接

基于图中A点的电位为虚地，略去流入运放的电流，则由图 1-1 得，

$$G(S) = \frac{-U_o}{U_i} = \frac{Z_2}{Z_1} \quad 1)$$

由上式可求得由下列模拟电路组成的典型环节的传递函数及其单位阶跃响应

1) 比例环节

比例环节的模拟电路如图 1-2 所示：

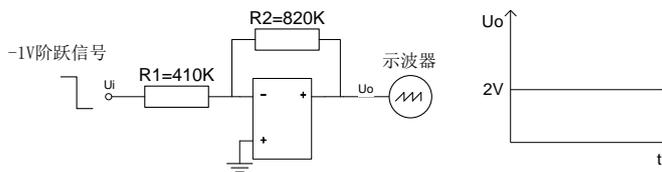


图 1-2 比例环节

$$G(S) = \frac{Z_2}{Z_1} = \frac{820K}{410K} = 2$$

2) 惯性环节

$$G(S) = \frac{Z_2}{Z_1} = \frac{\frac{R_2 / CS}{R_2 + 1 / CS}}{R_1} = \frac{R_2}{R_1} \cdot \frac{1}{R_2 CS + 1} = \frac{K}{TS + 1}$$

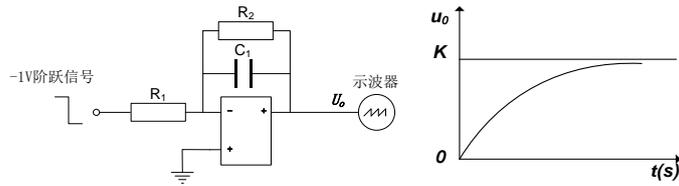


图 1-3 惯性环节

3) 积分环节

$$G(S) = \frac{Z_2}{Z_1} = \frac{1 / CS}{R} = \frac{1}{RCS} = \frac{1}{TS}$$

式中积分时间常数 $T = RC$

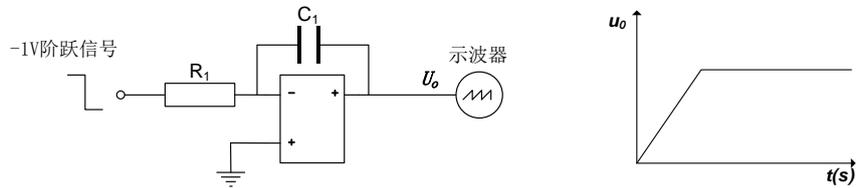


图 1-4 积分环节

4) 比例微分环节 (PD), 其电路图以及阶跃响应曲线如图 1-5 所示

$$G(S) = \frac{Z_2}{Z_1} = \frac{\frac{R_2}{R_1 / CS}}{R_1 + 1 / CS} = \frac{R_2}{R_1} \cdot (R_1 CS + 1) = K(T_D S + 1)$$

其中 $K = R_2 / R_1$, $T_D = R_1 C$

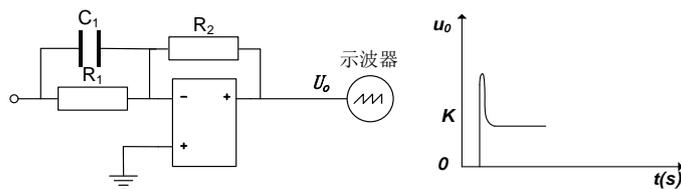


图 1-5 比例微分环节

5) 比例积分环节, 其接线图单位阶跃响应如图 1-6 所示

$$G(S) = \frac{Z_2}{Z_1} = \frac{R_2 + 1/CS}{R_1} = \frac{R_2CS + 1}{R_1CS} = \frac{R_2}{R_1} + \frac{1}{R_1CS} = \frac{R_2}{R_1} \cdot \left(1 + \frac{1}{R_2CS}\right) = K \left(1 + \frac{1}{T_2S}\right)$$

式中 $K = R_2/R_1$, $T_2 = R_2C$

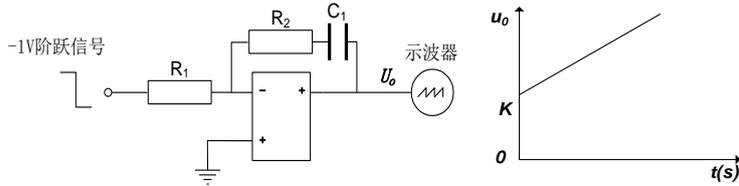


图 1-6 比例积分环节

6) 振荡环节, 其原理框图、接线图及单位阶跃响应分别如下所示。

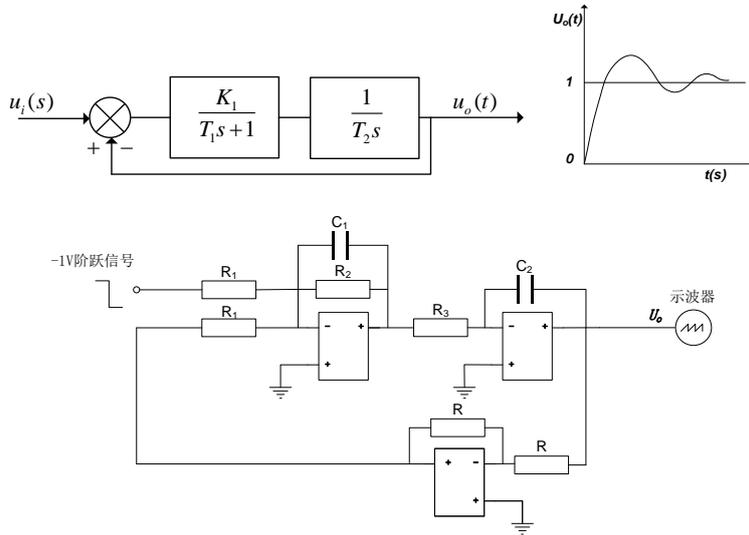


图 1-8 振荡环节接线图

图 1-8 为振荡环节的模拟线路图, 它是由惯性环节, 积分环节和一个反相器组成。根据他们的传递函数, 可以画出图 1-7 所示的方框图, 图中 $K = R_2/R_1$, $T_1 = R_2C_1$, $T_2 = R_3C_2$, 由图 1-8 可求得开环传递函数为:

$$G(S) = \frac{K}{S(T_1S + 1)}$$

其中 $K = \frac{K_1}{T}$, 则

$$\frac{U_o(S)}{U_i(S)} = \frac{K}{T_1S^2 + S + K} = \frac{K/T_1}{S^2 + S/T_1 + K/T_1} = \frac{\omega_n^2}{S^2 + 2\xi\omega_nS + \omega_n^2}$$

$$\omega_n = \sqrt{K/T_1}, 2\xi\omega_n = 1/T_1, \xi = 1/(2T_1\omega_n) = 1/(2\sqrt{KT_1})$$

欲使图 1-8 为振荡环节，须调整参数 K 和 T_1 ，使 $0 < \xi < 1$ ，呈欠阻尼状态，即环节的单位阶跃响应呈振荡衰减形式。

四、实验内容

分别画出比例、惯性、积分、微分和振荡环节的电子电路图。

按下列典型环节的传递函数，调节相应的模拟电路的参数，观察并记录其单位阶跃响应波形。

- 1、比例环节 $G_1(S) = 1$ 和 $G_2(S) = 1$
- 2、积分环节 $G_1(S) = 1/S$ 和 $G_2(S) = 1/0.5S$
- 3、比例微分环节 $G_1(S) = 2 + S$ 和 $G_2(S) = 1 + 2S$
- 4、惯性环节 $G_1(S) = \frac{1}{S+1}$ 和 $G_2(S) = \frac{1}{0.5S+1}$
- 5、比例积分环节 $G_1(S) = 1 + \frac{1}{S}$ 和 $G_2(S) = 2(1 + \frac{1}{2S})$
- 6、振荡环节 $G(S) = \frac{K}{T_1S^2+S+K} = \frac{10}{0.1S^2+S+10}$

五、实验注意事项

- 1、输入的单位阶跃信号取自实验箱中的阶跃信号发生器。
- 2、电子电路中的电阻取千欧，电容为微法。

六、实验报告要求

- 1、画出六种典型环节的实验电路图，并注明相应的参数。
- 2、画出各典型环节的单位阶跃响应波形，并分析参数对响应曲线的影响。
- 3、写出实验心得体会。

七、实验思考题

- 1、用运放模拟典型环节时，其传递函数是在哪两个假设条件下近似导出的？
- 2、积分环节和惯性环节主要差别是什么？在什么条件下，惯性环节可以近似地视为积分环节？在什么条件下，又可以视为比例环节？

3、如何根据阶跃响应的波形，确定积分环节和惯性环节的时间常数？

实验二 一阶系统的时域响应及参数测定

一、实验目的

- 1、观察一阶系统在单位阶跃和斜坡输入信号作用下的瞬态响应。
- 2、根据一阶系统的单位阶跃响应曲线确定一阶系统的时间常数。

二、实验仪器

- 1)、控制理论实验箱一台
- 2)、示波器一台
- 3)、万用表一只

三、实验原理

图 2-1 为一阶系统的模拟电路图。由该图可知 $i_0 = i_1 - i_2$ 即

$$\frac{u_i}{R_0} - \frac{u_o}{R_0} = -\frac{u_o}{1/CS}$$

$$\frac{\Delta u}{R_0} = -\frac{u_o}{1/CS}$$

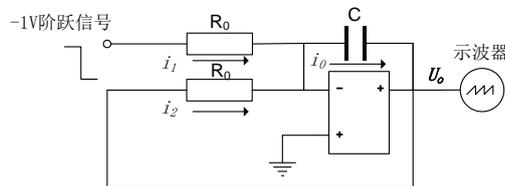


图 2-1 一阶系统模拟电路

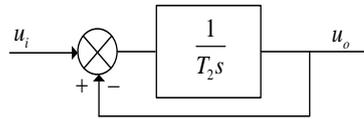


图 2-2 一阶系统原理图

根据上式，画出图 2-2 所示的方框图，其中 $T = R_0 C$ 。由图 2-2 得：

$$\frac{U_o(s)}{U_i(s)} = \frac{1}{TS+1}$$

令 $u_i(t) = 1(t)$ ，即 $U(S) = \frac{1}{S}$ ，则系统的输出为

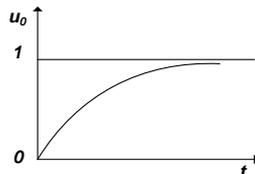


图 2-3 一阶系统单位阶跃响应

$$U_o(S) = \frac{1}{S(TS+1)} = \frac{1}{S} - \frac{1}{S+1/T}$$

取拉氏反变换得 $U_o(t) = 1 - e^{-\frac{1}{T}t}$ ，图 2-3 为一阶系统的单位阶跃响应曲线。

当 $t=T$ 时， $C(T) = 1 - e^{-1} = 0.632$ 。这表示当 $C(t)$ 上升到稳定值的 63.2% 时，对应的时间就是一阶系统的时间常数 T ，根据这个原理，由图 2-3 可测得一阶系统的时间常数 T ，由上式可知，系统的稳态值为 1，因而该系统的跟踪阶跃输入的稳态误差 $e_{ss} = 0$ 。

这表明一阶系统能跟踪斜坡信号输入，但有稳态误差存在。其误差的大小为系统的时间常数 T 。

四、实验内容

- 1、根据图 2-1 所示的模拟电路，调整 R_0 和 C 的值，使时间常数 $T = 1S$ 和 $T = 0.1S$ 。
- 2、 $u_i(t) = 1V$ 时，观察并记录一阶系统的时间常数 T 分别为 $1S$ 和 $0.1S$ 时的单位阶跃响应曲线，并标注时间坐标轴。
- 3、当 $u_i(t) = t$ 时，观察并记录一阶系统时间常数 T 为 $1S$ 和 $0.1S$ 时的响应曲线，其中斜坡信号可以通过实验箱中的三角波信号获得，或者把单位阶跃信号通过一个积分器获得。

五、实验报告

- 1、根据实验，画出一阶系统的时间常数 $T=1S$ 时的单位阶跃响应曲线，并由实测的曲线求得时间常数 T 。
- 2、观察并记录一阶系统的斜坡响应曲线，并由图确定跟踪误差 e_{ss} ，这一误差值与由终值定理求得的是否相等？分析产生误差的原因。

六、实验思考题

- 1、一阶系统为什么对阶跃输入的稳态误差为零，而对单位斜坡输入的稳态误差为 T ？
- 2、一阶系统的单位斜坡响应能否由其单位阶跃响应求得？试说明之。

实验三 二阶系统的瞬态响应分析

一、实验目的

- 1、熟悉二阶模拟系统的组成。
- 2、研究二阶系统分别工作在 $\xi = 1, 0 < \xi < 1$ 和 $\xi > 1$ 三种状态下的单位阶跃响应。
- 3、分析增益 K 对二阶系统单位阶跃响应的超调量 σ_p 、峰值时间 $t_p = 1$ 和调整时间 $t_s = 1$ 。
- 4、研究系统在不同 K 值时对斜坡输入的稳态跟踪误差。

二、实验仪器

- 1)、控制理论实验箱一台
- 2)、示波器一台
- 3)、万用表一只

三、实验原理

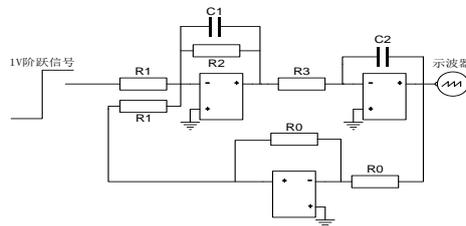


图 3-1 二阶系统模拟电路

图 3-1 为二阶系统的模拟电路图，它是由惯性环节、积分环节和反相器组成。

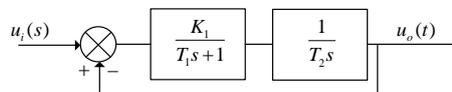


图 3-2 二阶系统原理框图

图 3-2 为图 3-1 的原理方框图，图中 $k = \frac{R_2}{R_1}, T_1 = R_2 C_1, T_2 = R_3 C_2$ 。

由图 3-2 求得二阶系统的闭环传递函数为：

$$\frac{U_o(S)}{U_i(S)} = \frac{K}{T_1 T_2 S^2 + T_2 S + K} = \frac{K / T_1 T_2}{S^2 + T_1 S + K / T_1 T_2} \quad (1)$$

而二阶系统标准传递函数为:

$$G(S) = \frac{\omega_n^2}{S^2 + 2\xi\omega_n S + \omega_n^2} \quad (2)$$

对比式 (1) 和式 (2), 得

$$\omega_n = \sqrt{K/T_1 T_2}, \quad \xi = \sqrt{T_2/4T_1 K}$$

若令 $T_1 = 0.2S, T_2 = 0.5S$, 则 $\omega_n = \sqrt{10K}, \xi = \sqrt{\frac{0.625}{K}}$

调节开环增益 K 值, 不仅能改变系统无阻尼自然振荡频率 ω_n 和 ξ 的值, 还可以得到过阻尼 ($\xi > 1$)、临界阻尼 ($\xi = 1$) 和欠阻尼 ($\xi < 1$) 三种情况下的阶跃响应曲线。

当 $K > 0.625, 0 < \xi < 1$, 系统处于欠阻尼状态, 它的单位阶跃响应表达式为

$$u_o(t) = 1 - \frac{1}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t} \sin(\omega_d t + \text{tg}^{-1} \frac{\sqrt{1-\xi^2}}{\xi}) \quad (3)$$

式中 $\omega_d = \omega_n \sqrt{1-\xi^2}$, 图 3-3 为二阶系统在欠阻尼状况下 d 单位阶跃响应曲线

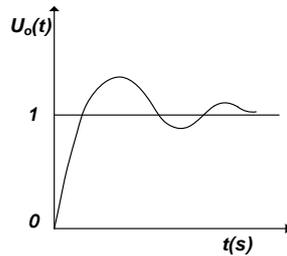


图 3-3 $0 < \xi < 1$ 时的阶跃响应曲线

(2) 当 $K = 0.625$ 时, $\xi = 1$, 系统处在临界阻尼状态, 它的单位阶跃响应表达式为

$$U_o(t) = 1 - (1 + \omega_n t) e^{(-\omega_n t)}$$

如图 3-4 为二阶系统工作临界阻尼时的单位响应曲线。

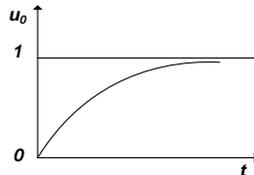


图 3-4 $\xi = 1$ 时的阶跃响应曲线

(3) 当 $K < 0.625$ 时, $\xi > 1$, 系统工作在过阻尼状态, 它的单位阶跃响应曲线和临界阻尼时的单位阶跃响应一样, 为单调的指数上升曲线, 但后者的上升速度比前者缓慢。

四、实验内容

1、根据图 3-1，调节相应的参数，使系统的开环传递函数为：

$$G(S) = \frac{K}{0.5S(0.2S+1)}$$

2、令 $u_i(t) = 1V$ 在示波器上观察不同 $K(K=10,5,2,0.5)$ 时的单位阶跃响应的波形，并由实验求得相应的波形，并由实验求得相应的 σ_p 、 t_p 和 t_s 的值。

3、调节开环增益 K ，使二阶系统的阻尼比 $\xi = \frac{1}{\sqrt{2}} = 0.707$ ，观察并记录此时的单位阶跃响应波形和 σ_p 、 t_p 和 t_s 的值。

4、观察并记录在不同 K 值时，系统跟踪斜坡信号的稳态误差。

五、实验报告

1、画出二阶系统在不同 K 值（10，5，2，0.5）下的 4 条瞬态响应曲线，并注明时间坐标轴。

2、按图 3-2 所示的二阶系统，计算 $K=0.625$ ， $K=1$ 和 $K=0.312$ 三种情况下， ξ 和 ω_n 值。据此，求得相应的动态性能指标 σ_p 、 t_p 和 t_s ，并与实验所得出结果作一比较。

3、写出本实验的心得与体会。

六、实验思考题

1、如果阶跃输入信号的幅值过大，会在实验中产生什么后果？

2、在电子模拟系统中，如何实现负反馈和单位负反馈？

3、为什么本实验的模拟系统中要用三只运算放大器？

实验四 三阶系统的瞬态响应及稳定性分析

一、实验目的

- 1)、掌握三阶系统的模拟电路图。
- 2)、由实验证明开环增益 K 对三阶系统的动态性能和稳定性的影响。
- 3)、研究时间常数 T 对三阶系统稳定性的影响。

二、实验仪器

- 1)、控制理论实验箱一台
- 2)、示波器一台
- 3)、万用表一只

实验原理

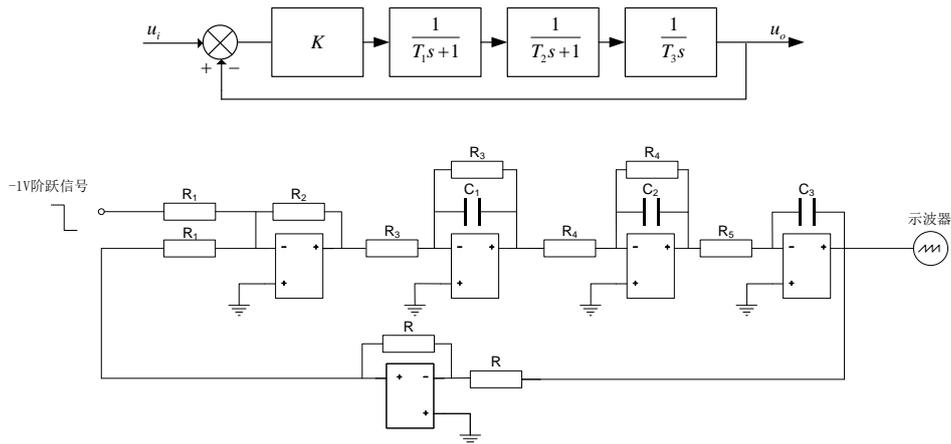


图 4-1 为三阶系统的方框图，它的模拟电路如图 4-2 所示，它的闭环传递函数为：

$$\frac{U_o(S)}{U_i(S)} = \frac{K}{T_3S(T_1S+1)(T_2S+1)+K}$$

该系统特征方程为

$$T_1T_2T_3S^3 + T_3(T_1+T_2)S^2 + T_3S + K = 0$$

其中 $K = \frac{R_2}{R_1}$, $T_1 = R_3C_1$, $T_2 = R_4C_2$

若令 $T_1 = 0.2S$, $T_2 = 0.1S$, $T_3 = 0.5S$

则上式改写为

$$S^3 + 15S^2 + 50S + 100K = 0$$

用劳斯稳定判据，求得该系统的临界稳定性增益 $K = 7.5$ 。这表示 $K > 7.5$ 时，系统为不稳定， $K < 7.5$ 时，系统才能稳定运行， $K = 7.5$ 时，系统作等幅振荡。

除了开环增益 K 对系统的动态性能和稳态性有影响外，系统中任何一个时间常数的变化对系统的稳定性都有影响，对此说明如下：

令系统剪切频率为 ω_c ，则在该频率时的开环频率特性的相位为：

$$\varphi(\omega_c) = -90^\circ - \tan^{-1}T_1\omega_c - \tan^{-1}T_2\omega_c$$

$$\text{相位裕量 } \gamma = 180^\circ + \varphi(\omega_c) = 90^\circ - \tan^{-1}T_1\omega_c - \tan^{-1}T_2\omega_c$$

由上式可见，时间常数 T_1 和 T_2 的增大都会使 γ 减小。

四、实验内容与步骤

图 4-1 所示的三阶系统开环传递函数为

$$G(s) = \frac{K}{T_3s(T_1s + 1)(T_2s + 1)}$$

- 1)、按 $K = 10$, $T_1 = 0.2S$, $T_2 = 0.05S$, $T_3 = 0.5S$ 的要求，调整图 4-2 中的相应参数。
- 2)、用慢扫描示波器观察并记录三阶系统单位阶跃响应曲线。
- 3)、令 $T_1 = 0.2S$, $T_2 = 0.1S$, $T_3 = 0.5S$ 用示波器观察并记录 K 分别为 5, 7.5 和 10 三种情况下的单位阶跃响应曲线。
- 4)、令 $K = 10$, $T_1 = 0.2S$, $T_3 = 0.5S$, 用示波器观察并记录 T_2 分别为 0.1S 和 0.5S 时的单位阶跃响应曲线。

五、实验报告

- 1)、作出 $K = 5$ 、7.5 和 10 三种情况下的单位阶跃响应波形图，据此分析 K 的变化对系统动态性能和稳定性的影响。
- 2)、作出 $K = 10$, $T_1 = 0.2S$, $T_3 = 0.5S$ T_2 分别为 0.1S 和 0.5S 时的单位阶跃响应波形图，并分析时间常数 T_2 的变化对系统稳定性的影响。
- 3)、写出本实验的心得与体会。

六、实验思考题

- 1)、为使系统能稳定地工作，开环增益应适当取小还是取大？

- 2)、系统中的小惯性环节和大惯性环节哪个对系统稳定性的影响大,为什么?
- 3)、试解释在三阶系统的实验中,输出为什么会出出现削顶的等幅振荡?
- 4)、为什么图 3-2 和图 4-1 所示的二阶系统与三阶系统对阶跃输入信号的稳态误差都为零?
- 5)、为什么在二阶系统和三阶系统的模拟电路中所用运算放大器都为奇数个?

实验五 频率特性的测量

一、实验目的

1. 掌握用李沙育图形法，测量各典型环节的频率特性；
2. 根据所测得的频率特性，作出伯德图，据此求得环节的传递函数。

二、实验仪器设备

1. 控制理论实验箱一台；
2. 示波器一台；
3. 任意函数信号发生器一台；
4. 万用表一只。

三、实验内容与测量方法

对于稳定的线性定常系统或环节，当其输入端加入一正弦信号 $X(t) = X_m \sin \omega t$ ，它的稳态输出是一与输入信号同频率的正弦信号，但其幅值和相位随着输入信号频率 ω 的改变而改变。输出信号为

$$Y(t) = Y_m \sin \omega t + \varphi = Y_m |G(j\omega)| \sin(\omega t + \varphi)$$

其中 $|G(j\omega)| = \frac{Y_m}{X_m}$ ， $\varphi = \operatorname{arctan}(j\omega)$

只要改变输入信号的频率，就可以测得输出信号与输入信号的幅值比 $|G(j\omega)|$ 和它们的相位差 $\varphi(\omega)$ 。不断改变 $x(t)$ 的频率，就可测得被测环节（系统）的幅频特性和相频特性。

本实验采用李沙育图形法，图 5-1 为测试的方框图

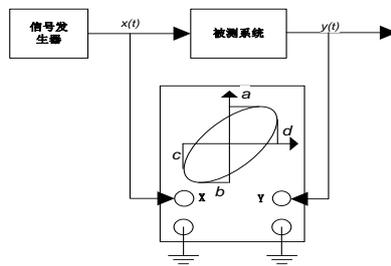


图 5-1 频率特性测试方框图

在表（1）中列出了超前于滞后时相位的计算公式和光点的转向。

相角 ϕ	超前		滞后	
	$0^\circ \sim 90^\circ$	$90^\circ \sim 180^\circ$	$0^\circ \sim 90^\circ$	$90^\circ \sim 180^\circ$
图形				
计算公式	$\varphi = \sin^{-1} \frac{2Y_0}{2Y_m}$ $= \sin^{-1} \frac{2X_0}{2X_m}$	$\varphi = 180^\circ - \sin^{-1} \frac{2Y_0}{2Y_m}$ $= 180^\circ - \sin^{-1} \frac{2X_0}{2X_m}$	$\varphi = \sin^{-1} \frac{2Y_0}{2Y_m}$ $= \sin^{-1} \frac{2X_0}{2X_m}$	$\varphi = 180^\circ - \sin^{-1} \frac{2Y_0}{2Y_m}$ $= 180^\circ - \sin^{-1} \frac{2X_0}{2X_m}$
光点转向	顺时针	顺时针	逆时针	逆时针

表中 $2Y_0$ 为椭圆与 Y 轴交点之间的长度， $2X_0$ 为椭圆与 X 轴交点之间的距离， X_m 和 Y_m 分别为 $X(t)$ 和 $Y(t)$ 的幅值。

四、实验内容

1、R-C 网络的频率特性。图 5-2 为接线图，分别测试其幅频特性和相频特性。

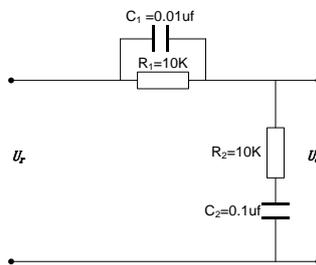


图 5-2 滞后--超前校正网络的接线图

2、闭环频率特性的测试。被测的二阶系统如图 5-3 所示，图 5-4 为它的模拟电路图。

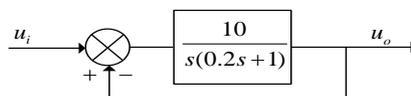


图 5-3 二阶控制系统

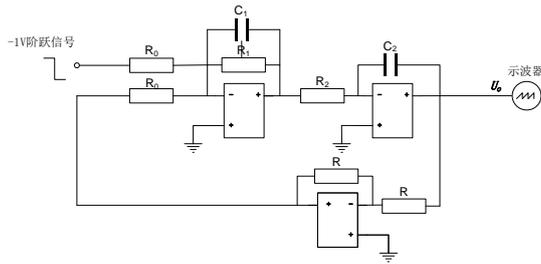


图 5-4 被测二阶系统的接线图

相频特性的测试数据

$\omega(\text{rad/s})$	f	$2X_m$	$2X_0$	实测 $\varphi(\omega)$	计算 $\varphi(\omega)$	光点转动方向

其中 $\varphi(\omega) = \sin^{-1}\left(\frac{X_0}{X_m}\right)$

幅频特性的测试数据

$\omega(\text{rad/s})$	f	$2Y_m$	$2X_m$	$2Y_m/2X_m$	实测 $L(\omega)$	计算 $L(\omega)$

其中 $L(\omega) = 20\lg\left(\frac{2Y_m}{2X_m}\right)$

注意事项

- 1、输入信号的频率的取值范围
- 2、在做闭环幅频特性时，在幅值最大值附近多取几组值，以正确确定谐振峰值 M_r 和频率 ω_r 。

五、实验报告

- 1、测试滞后超前网络的相关数据，填入表中。
- 2、按测得的实验数据，分别画出 $\varphi(\omega) \sim \omega$ 和 $20\lg |G(j\omega)| \sim \omega$ 的曲线。
- 3、将由实验求得的传递函数与理论的 $G(s)$ 作一比较，并分析产生误差的原因。
- 4、根据实验作出二阶系统闭环幅频特性曲线，由图求系统的谐振频率 ω_r 和谐振峰值 M_r ，并于理论计算结果进行比较。

六、实验思考题

- 1、在实验中如何选择输入正弦信号的幅值？
- 2、测试频率特性时，示波器 Y 轴输入开关为什么选择直流？

3、测试相频特性时，若把信号发生器的正弦信号送入 Y 轴，被测系统的输出信号送入 X 轴，则根据椭圆光点的转动方向，如何确定相位的超前和迟后？

实验六 线性定常系统的串联校正

一、实验目的

1. 通过实验，理解所加校正装置的结构、特性和对系统性能的影响；
2. 掌握串联校正常用设计方法和对系统的实时调试技术。

二、实验设备

1. 控制理论实验箱一台；
2. 示波器一台；
3. 万用表一只。

三、实验原理

图 6-1 为一加串联校正后系统的方框图。图中校正装置 $G_c(S)$ 与被控对象 $G_0(S)$ 串联连接。

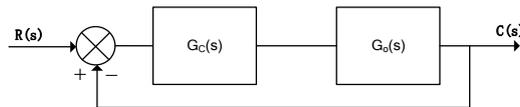


图 6-1 加串联校正后系统的方框图

串联校正有以下三种形式：

- 1) 超前校正，这种校正利用超前校正装置的相位超前特性来改善系统的动态性能。
- 2) 滞后校正，这种校正利用滞后校正装置的高频幅值衰减特性，使系统在满足稳态性能的前提下又能满足其动态性能的要求。

3) 滞后超前校正，由于这种校正既有超前校正的特点，又有滞后校正的优点。因而它适用系统需要同时改善稳态和动态性能的场所。校正装置有无源和有源二种。基于后者与被控对象相连接时，不存在着负载效应，故得到广泛地应用。

常用的校正方法：零极点对消法（时域法；采用超前校正）。

所谓零极点对消法就是使校正变量 $G_c(S)$ 中的零点抵消被控对象 $G_0(s)$ 中不希望的极点，以使系统的动、静态性能均能满足设计要求。设校正前系统的方框图如图 6-2 所示。

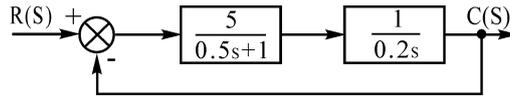


图 6-2 二阶闭环系统的方框图

1、性能要求

静态速度误差系数： $K_v = 25 \frac{1}{s}$ ，超调量： $\delta_p \leq 0.2$ ；上升时间： $t_s \leq 1S$ 。

2、校正前系统的性能分析

校正前系统的开环传递函数为：

$$G_0(S) = \frac{5}{0.2S(0.5S+1)} = \frac{25}{S(0.5S+1)}$$

系统的速度误差系数为： $K_v = \lim_{s \rightarrow 0} sG_0(S) = 25$ ，刚好满足稳态的要求。根据系统的闭环传

递函数

$$\Phi(S) = \frac{G_0(S)}{1+G_0(S)} = \frac{50}{S^2 + 2S + 50} = \frac{\omega_n^2}{S^2 + 2\zeta\omega_n S + \omega_n^2}$$

求得 $\omega_n = \sqrt{50}$ ， $2\zeta\omega_n = 2$ ， $\zeta = \frac{1}{\omega_n} = \frac{1}{\sqrt{50}} = 0.14$

代入二阶系统超调量 δ_p 的计算公式，即可确定该系统的超调量 δ_p ，即 $\delta_p = e^{-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}} =$

$$0.63, t_s = \frac{3}{\zeta\omega_n} = 3S(\Delta = \pm 0.05)$$

这表明当系统满足稳态性能指标 K_p 的要求后，其动态性能距设计要求甚远。为此，必须在系统中加一合适的校正装置，以使校正后系统的性能同时满足稳态和动态性能指标的要求。

3、校正装置的设计

根据对校正后系统的性能指标要求，确定系统的 ζ 和 ω_n 。即由 $\delta_p \leq 0.2 = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}}$

求得 $\xi \geq 0.05$

$$t_s \doteq \frac{3}{\xi\omega_n} \leq 1S(\Delta = \pm 0.05), \text{ 解得 } \omega_n \geq \frac{3}{0.5} = 6$$

根据零极点对消法则，令校正装置的传递函数

$$G_C(s) = \frac{0.5S + 1}{TS + 1}$$

则校正后系统的开环传递函数为：

$$G(S) = G_c(S)G_o(S) = \frac{0.5S + 1}{TS + 1} \times \frac{25}{S(0.5S + 1)} = \frac{25}{S(TS + 1)}$$

相应的闭环传递函数

$$\phi(S) = \frac{G(S)}{G(S) + 1} = \frac{25}{TS^2 + S + 25} = \frac{25/T}{S^2 + S/T + 25/T} = \frac{\omega_n^2}{S^2 + 2\xi\omega_n S + \omega_n^2}$$

于是有： $\omega_n^2 = \frac{25}{T}$ ， $2\xi\omega_n = \frac{1}{T}$

为使校正后系统的超调量 $\delta_p \leq 20\%$ ，这里取 $\xi = 0.5$ ($\delta_p \approx 16.3\%$)，则 $2 \times 0.5 \sqrt{\frac{25}{T}} = \frac{1}{T}$ ，

$T = 0.04^s$ 。

这样所求校正装置的传递函数为：

$$G_o(S) = \frac{0.5S + 1}{0.04S + 1}$$

设校正装置 $G_c(S)$ 的模拟电路如图 6-3 所示。

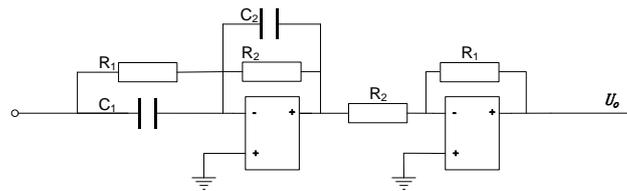
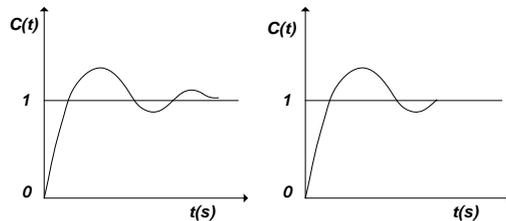


图 6-3 校正装置的电路图

其中 $R_1 = 510K$ ， $C_1 = 1\mu F$ ， $R_2 = 390K$ ， $C_2 = 0.1\mu F$ 时有

$$G_o(S) = \frac{R_1 C_1 S + 1}{R_2 C_2 S + 1} = \frac{0.51S + 1}{0.039S + 1} \approx \frac{0.5S + 1}{0.04S + 1}$$

图 6-4 (a)、(b) 分别为二阶系统校正前、后系统的单位阶跃响应的示意曲线。



(a) (δ_p 约为 63%)

(b) (δ_p 约为 16.3%)

图 6-4 加校正装置前后二阶系统的阶跃响应曲线

五、实验步骤

零极点对消法（时域法）进行串联校正

1、校正前

根据图 6-2 二阶系统的方框图，选择实验台上的通用电路单元设计并组建相应的模拟电路，如图 6-5 所示。

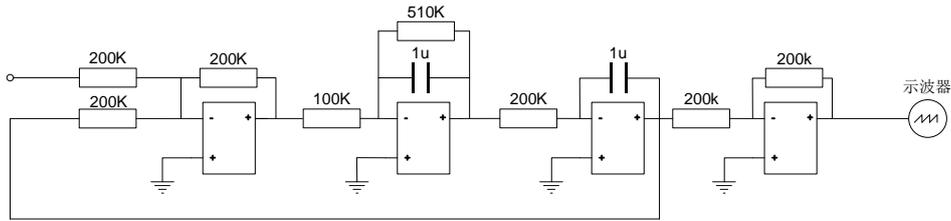


图 6-5 二阶闭环系统的模拟电路图(时域法)

在 r 输入端输入一个单位阶跃信号，观测并记录相应的实验曲线，并与理论值进行比较。

2、校正后

在图 6-5 的基础上加上一个串联校正装置在系统输入端输入一个单位阶跃信号，观测并记录相应的实验曲线，并与理论值进行比较，观测 δ_p 是否满足设计要求。

六、实验报告要求

1. 根据对系统性能的要求，设计系统的串联校正装置，并画出它的电路图；
2. 根据实验结果，画出校正前系统的阶跃响应曲线及相应的动态性能指标；
3. 观测引入校正装置后系统的阶跃响应曲线，并将由实验测得的性能指标与理论计算值作比较；
4. 实时调整校正装置的相关参数，使系统的动、静态性能均满足设计要求，并分析相应参数的改变对系统性能的影响。

七、实验思考题

1. 加入超前校正装置后，为什么系统的瞬态响应会变快？
2. 什么是超前校正装置和滞后校正装置，它们各利用校正装置的什么特性对系统进行校正？

第二部分 仿真环境简明介绍

§1 MATLAB

§ 1.1 MATLAB 简介

在科学研究和工程应用的过程中，往往需要进行大量的数学计算，传统的纸笔和计算器已经根本不能满足海量计算的要求。一些技术人员尝试用 Basic、Fortran 以及 C 语言编制程序来减轻计算的工作量。但编制程序不仅需要掌握所用语言的语法，还需要对有关算法进行深入分析，致使科学工作者在编程上花费过多时间。为了满足用户对工程数学计算的要求，一些软件公司相继推出了一批数学类科技应用软件，如 MATLAB、Xmath、Mathematica、Maple 等。其中，Mathworks 公司推出的 MATLAB（即 matrix 和 laboratory 的前三位字母组合）以其强大的功能和易用性受到越来越多的科技工作者的欢迎。

MATLAB 起初是作为矩阵实验室(Matrix Laboratory)提供对 LINPACK 和 EISPACK 矩阵软件包接口 (LINPACK 是解线性方程的 Fortran 链接库，EISPACK 是解特征值问题的 Fortran 链接库)。与 Basic、Fortran 以及 C 语言比较，MATLAB 的语法规则更简单，编程特点更贴近人的思维方式，在工程计算方面的编程效率远远高于其它编程语言。

Matlab 由主程序、Simulink 动态仿真系统、和 Matlab 工具箱三部分组成。其中主程序包括 Matlab 语言、工作环境以及应用程序；Simulink 动态仿真系统是一个相互交互的系统，用户制作一个模拟系统，并动态控制它；而工具箱就是 Matlab 基本语句的各种子程序和函数库。它可以分为功能性和学科性工具箱。功能性的工具箱主要用于扩展 Matlab 的符号计算功能、图形建模功能、文字处理功能和与硬件的实时交互过程，如符号计算工具箱等；学科性的工具箱则有较强的专业性，用于解决特定的问题，如信号处理工具箱和通信工具箱。

如今，Matlab 软件正在成为对数值、线性代数以及其他一些高等应用数学课程进行辅助教学的有力工具；在工程技术界，Matlab 软件也被用来构建与分析一些实际课题的数学模型，其典型的应用包括数值计算、算法预设计与验证，以及一些特殊矩阵的计算应用，如统计、图像处理、自动控制理论、数字信号处理、系统识别和神经网络等。它包括了被称作工具箱(Toolbox)的各类应用问题的求解工具。工具箱实际上是对 Matlab 软件进行扩展应用的一系列 Matlab 函数(称为 M 函数文件)，它可用来求解许多学科门类的数据处理与分析问题。

Matlab 的主要特点有：

1. Matlab 的单位为矩阵，它的表达式与数学工程计算类似，并且可以随时修改，而且行和列无需定义。
2. Matlab 对错误可以立即反映，并且进行解释，大大减少了调试和编程的工作量，且具有友好的人机界面。
3. Matlab 语言规则适于交流，而且具有强大的作图和数据可视化功能，并且简单、直观、方便。
4. Matlab 软件包有许多工具箱，各种工具箱能针对某一专门的问题或新算法解决，还为其它语言提供了各种接口，以实现数据的传递和共享。

1.2 MATLAB 功能介绍

以 2019a 版本为例。

1.matlab 界面

左上角，home（主页）标签下，找到 layout（布局）进行设置/复位，可以设置各板块的显示与隐藏。其中有几个部分，请务必必要显示



①Current Folder（当前文件夹）：设置为具有读写权限的文件夹，例如我的文档下建立一个 matlab 文件夹

②Command Window（命令窗口）：用来运行代码，所有的代码都是在这里输入

③Workspace（工作空间）：暂存所有运行结果的地方，关闭 matlab 后数据丢失

2、软件中的基本概念

2.1 函数

Matlab 不仅提供大量的函数，也可以建立自定义函数，方法是：Home->New->function（主页->新建->函数）。自定义函数一般保存在工作路径下。函数文件的特征是：扩展名 m，内容的第一行以 function 开头，后续内容是“输出变量=函数名(输入变量)”。且函数名和文件名相同。

函数在 Command Window 中运行，用来完成特定的计算任务，运行方式是输入“输出变量=函数名(输入变量)”，然后按回车。例如有个系统自带的函数是用来求绝对值的，函数名 abs，所以在 Command Window 里输入“a=abs(-1)”，就会显示运算结果为“a=1”。且运算结果

会在 **Workspace** 里出现一个变量 **a**，双击后可看到 **a** 的值是 1。

2.2 脚本

可理解为特殊的函数，函数内容的开头没有 **function**，因此没有输入、输出变量，也没有函数名。文件扩展名和函数一样是 **m**，也需要在 **Command Window** 里运行。脚本都是用户建立的，方法是：**Home->New Script**（主页->脚本）。一般保存在工作路径下。脚本的功能就是完成用户需要的、复杂的计算任务，通常脚本里会调用很多函数。

3.获得帮助

常用的获得帮助有四种方法

1)、**home** 标签里，有个 **Help** 标志，点开后可以获得各工具箱/产品的完整帮助文档。新版本中默认使用在线，改用本地帮助的办法是在 **home** 标签里，**Preferences** 下的 **matlab/Help** 里选择 **installed locally**

2)、**cn.mathworks.com** 官网上找到支持，然后可以获得教程。这种方法获得的帮助文档和第一种方法一样。

3)、在 **Command Window** 里输入 **doc+函数名** 来获得帮助。比如输入"**doc fft**"可以获得离散傅里叶变换函数 **fft** 的帮助和范例。这种方法获得的文档是前两种方法文档中的部分。当然，前提是你要知道函数名，才能找到帮助。这种方法适合于获得系统自带函数的使用说明。

4)、对于一个 **MATLAB** 的初学者来说，通过 **MATLAB** 提供的例子学习 **MATLAB** 语言也许是一种最快捷的方式了。打开 **MATLAB** 的例子演示方法。

a. 在 **MATLAB** 的命令窗口输入：**demo** 命令

b. 在 **MATLAB** 的帮助菜单执行“**Examples and Demos**”命令。

这二种方法都将在 **MATLAB** 命令窗口打开 **demo** 命令。

这几种方法中，最常用的是第三种，只要知道自己需要的函数名，就可以用这种方式获得说明和范例。

(二): 变量类型与赋值

matlab 提供的变量类型很多, 这里仅说明最基础的变量类型: 数值变量、符号变量。

1.数值变量

matlab 中所有的数值变量都是矩阵, 赋值时, 以方括号作为开头和结尾, 以英文逗号或空格分割同行元素, 以英文分号分割各列。例如在 Command Window 里输入

```
>> a=[1 2;3 4]
```

运算结果为:

```
a =  
    1     2  
    3     4
```

a 是一个数值变量。同时 workspace 里出现一个田字形的变量 a, 说明变量 a 的类型是数值型。

向量和数字可以视为特殊的矩阵, 例如

```
>> a=[1 2]
```

输出为:

```
a =  
    1     2
```

输入:

```
>> a=[1;2]
```

输出为:

```
a =  
    1  
    2
```

分别是行向量和列向量。

而

```
>> a=[1]
```

输出为:

```
a =  
    1
```

与

```
>> a=1
```

输出相同。

数值变量的命名要求是英文字母开头，不能包含特殊符号，大小写敏感。推荐采用下划线来进行分割，例如 `value_of_A`，与其他编程语言的命名规则相当。

赋值中，有时需要用到等差数列，例如定义一个向量 `a=[1 2 3]`，如果比较长，则可以使用

```
>> a=[1:1:3]
```

这里两个冒号的意思是起始值:步长:终值。采用这种赋值方式时可以获得一个等差数列行向量，并可以省略两侧的方括号。当步长为 1 时，可以省略步长和一个冒号，于是可以简写为

```
>> a=1:3
```

分块矩阵的表示方法如下:

```
>> a=[1 2;3 4]
```

输出为:

```
a =  
  
     1     2  
  
     3     4
```

定义变量 a 之后，

```
>> a=[1 2;3 4];  
  
>> b=a(1,2)
```

就可以把 a 的第一行第二列元素赋值给 b，当然也可以用

```
>> a(1,2)=3
```

来修改矩阵中部分元素的值。这里需要注意，序号必须是自然数，且不能是零。

2.符号变量

常用的赋值命令是

```
>> syms a b
```

syms 表示这里要定义一些符号变量，a 和 b 是变量名，符号变量的命名规则和数值变量一样。有时候也采用

```
>> syms a real
```

来强调 a 是实数变量，具体可以 doc syms 来获得帮助。

若变量之间存在依赖关系，此时可以定义

```
>> syms x y(x)
```

声明 x 是一个符号变量，同时声明 y 是一个符号变量，且 y 的值由 x 决定，这相当于数学中函数的概念。当然具体的函数关系并没有明确规定。也可以

```
>> syms x y z(x,y)
```

来定义符号变量 z , z 依赖 x 和 y 。这相当于二元函数的概念。这里的圆括号显然和数值变量中的圆括号含义完全不同, 同一个符号, 由于变量类型不同会有完全不同的含义。所以在学习 matlab 的过程中, 一定要区分数值变量和符号变量。

(三): 常用的运算符

算术运算符		关系运算符		逻辑运算符		操作符	
操作符	解释	操作符	解释	运算符	解释	符号	解释
+	加	==	等于	&	逻辑和	:	冒号, 很重要
-	减	~=	不等于		逻辑或	;	分隔行
*	矩阵乘	>	大于	~	逻辑非	,	分隔列
.*	数组乘	<	小于			()	用于运算次序
^	矩阵乘方	>=	大于等于			[]	构成向量和矩阵
.^	数组乘方	<=	小于等于			{}	构成单元数组
\	矩阵左除					.	小数点
.\	数组左除					..	父目录
/	矩阵右除					...	表示该项未完
./	数组右除					%	用于注释
						!	调用操作系统命令
						=	赋值

(四): 判断与流程控制

判断常用于数学中的分段问题, 更为复杂的问题则需要流程控制。具体如下

1. 逻辑变量

1.1 逻辑变量的赋值

有些语言中如 C 语言中, 逻辑变量是一种专门的变量类型, 其值为 true 或者 false, matlab 中也有这个类型, 但其值为 1 或者 0。赋值方式是“变量名=表达式”, 变量名的命名规则与数值变量一样, 表达式是一种判断, 比如数值大小的判断, 或者高级函数的判断。例如输入

```
>> a=1>2
```

输出为:

```
a =  
  
logical  
  
0
```

定义一个逻辑变量 a，其值是 0，表示 false，在 workspace 里可以看见 a 的类型是 logical。

对初学者，可以认为数值大小的判断，是指两个 1*1 维的数值变量的比较。数值大小比较的方法有：>大于；<小于；>=大于等于；<=小于等于；==等于；~=不等于 6 种。因为数值计算有舍入误差，所以等于这个判断，有时需要用

```
>> abs(x1-x2)<eps
```

这样的不等式来代替，其中 eps 是 matlab 中最小的非零数值，相当于舍入误差的标准。高级函数的判断，是指系统自带的一些函数，例如

```
>> a=exist(x)
```

用来判断是否存在变量、函数或者文件。

1.2 逻辑变量的逻辑运算

逻辑变量的取值只有 0 或者 1，他们之间可以进行逻辑运算，运算符有：&&与；||或；~非。这里需要注意：

- 1) ~单独使用表示非运算，要区别于不等于~=的连用
- 2) 如果是高级函数的判断配合非运算，也可以实现判断，例如

```
>> a=~isprime(x)
```

就相当于判断 x 是否为合数

1.3 逻辑变量的数值运算

matlab 中逻辑变量可以参与数值运算，参与时，会被数值 0 和 1 代替。例如

```
>> a=1>2;b=a+1
```

输出为:

```
b =  
  
    1
```

matlab 会判断正在进行数值运算，然后将逻辑变量 a 转化成数值变量，由于 a 在逻辑变量时表示 false，所以转化时，a 相当于数值变量 0，因此会得到数值变量 $b=0+1=1$ 。

2.流程控制

matlab 的流程控制和 C 语言类似，if 和 switch 作为判断依据，相当于流程图的菱形框，for 和 while 作为循环，相当于流程图的反向箭头。

2.1 if 判断

语法是:

```
if 逻辑变量 1  
    执行语句 1  
elseif 逻辑变量 2  
    执行语句 2  
else  
    执行语句 3  
End
```

其含义与 C 语言中一致，是顺序判断，即逐个 if 或者 elseif 进行判断，若逻辑变量为 1，则执行语句，若为 0 则进行下一个判断。其中逻辑变量经常用多个逻辑变量的逻辑运算结果来代替，例如我们经常看到的是

```
>> if x>1 && y>1
```

与

```
>> a=(x>1) && (y>1)

>> if a
```

是等效的。有时候为了书写和注释，也会先定义一个逻辑变量 flag，然后再 if flag。

2.2 switch 判断

switch 的语法是：

```
switch 数值变量 1

case 数值 1
    执行语句 1

case 数值 2
    执行语句 2

otherwise
    执行语句 3

End
```

switch 与 if 最大的不同是，switch 只进行一次判断，即根据数值变量 1 的值来决定执行哪一个 case，而不像 if 会判断多个 elseif。需要强调的是，只有当数值变量 1==数值 1 时，才会执行语句 1，而之前说过，==这个判断是有舍入误差的，因此 switch 一般用于数值变量 1 只取整数时，比如 sfun 的判断。对初学者而言，也可以先无视 switch。

2.3 for 循环

for 循环的语法是：

```
for 数值变量=向量

    执行语句

End
```

其含义是，数值变量会取向量中的每一个值，然后执行语句，在执行语句中，数值变量是一个 1*1 的矩阵。比如

```
>> for a=[0 1 3]
```

则会让 a=0, 执行语句, 然后 a=1, 执行语句, 最后 a=3, 执行语句。这里容易和赋值语句混淆, 因为 a=[0 1 3]是一个数值变量赋值, 赋值后 a 是个向量, 但加上了 for 后, a 是 1*1 矩阵。另外, 在数值变量赋值那段, 我们说过有一种常用的赋值方法是“a=初始值:步长:终值”, 其结果是把 a 赋值成一个向量, 因此这也广泛用于 for 语句中

```
>> for a=初始值:步长:终值
```

则在执行语句中, a 是一个遍历初值到终值的 1*1 矩阵。

2.4 while 循环

while 语句的语法是:

```
while 逻辑变量
    执行语句
End
```

其含义是, 当逻辑变量为 1 时, 执行语句, 直到逻辑变量为 0, 因此 while 的执行语句中, 一定有改变逻辑变量的部分, 否则就是死循环。比如写了 while a<10, 就一定会在执行语句中有类似 a=a+1 这样的赋值语句, 使得若干次循环后, 逻辑变量为 0。总体而言, while 并不如 for 稳定, 所以初学者也可以无视 while。

(五): 编程习惯

前言: matlab 的基本使用方法差不多介绍完了, 确定问题类型 (数值/符号), 编程 (函数/脚本) 并运行即可。但具体编程过程中, 良好的习惯非常重要, 一方面便于调试, 另一方面便于代码维护和升级。

1、注释

matlab 提供两种注释, 分别是%%和%。%%一般独占一行, 用来分段, 两个%%之间的内容称为一段, 在程序调试时, 可以设置为运行一段代码后暂停, 以便查看一段代码的运行结果。使用范例如下:

```
%%赋值
赋值代码
%%第一阶段计算
```

```
第一阶段代码  
%%第二阶段计算  
第二阶段代码  
%%输出
```

输出阶段代码这样在调试时，若设置分段运行，则运行完赋值代码后，程序自动进入调试模式，此时可以在 workspace 里看到赋值段的运行结果，以便确认这段代码是否正确。%常用于一般代码的末尾，用来说明这一行代码的含义，例如

```
g=9.8%重力加速度赋值
```

这样之后调试时，可以快速找到这一行，并修改相应代码。需要注意的是，%仅在行内，对其后的内容有效，遇到回车换行后失效。此外，无论是%%还是%，注释内容会显示为绿色。

2、分行

无论 matlab 还是其他程序语言，也无论函数还是脚本，依次逐行运行是基本特征，因此我们一般不希望一行的内容太多，因为这样不方便差错。但如果一定要使用这样一行很长的代码，matlab 提供了一种分行的方式。使用方法是在代码中输入三个.，然后回车。这样产生的代码比如：

```
a=...  
1
```

这行代码和 $a=1$ 完全一样。需要指出的是，这种分行方式仅在输入时有效，在实际运行中，matlab 仍然认为这是一行代码。这种分行的另一种用法是用在矩阵赋值中，比如

```
a=[1 2;...  
3 4]
```

这样可以很明显看出 a 是 2×2 矩阵。

3、分号

matlab 的每一行代码，一般都会在 command window 里显示运行结果，如果不想显示，可以在代码后面写一个分号；，这样就可以不显示该行的运行结果。需要注意的是：其一，写

不写分号不影响程序的运行和结果；其二，显示运行结果是需要占用计算时间的，因此一般的语句都会写上分号；其三，if、for 等流程控制语句，这一行不加分号。

4、先定义再使用

虽然 matlab 中可以直接使用 `a(2,2)=1` 这样的语句，而无需先定义 `a` 是多大维度的矩阵，但这样可能造成运行变慢。先定义再使用依然是个好习惯，尤其是赋值语句中有讲过 `zeros` 和 `ones` 这样的语句，先把矩阵的维度定义好，再逐个元素赋值或者局部赋值，有助提高运行效率。

5、特殊运算代替循环

matlab 相较于 C 等语言的强大之处在于用特殊运算代替循环，比如 `a` 和 `b` 是同纬度矩阵，要将他们对应的元素相乘。在常用运算中，我们讲过，在 matlab 中可以用点乘 `.*`。在 C 语言中则需要一个 for 循环，遍历 `a` 和 `b` 的每个元素，相乘后赋值给结果矩阵。当然，matlab 中也可以用 for 循环来实现这个操作，但相比特殊运算，for 循环的效率实在是差太多。特殊运算代替循环也成为了提高运行效率的主要方法。

(六)：编程调试与常见报错

1、打断点和分段

写完脚本并保存后，在每一行可执行代码前，有一个行号，和一个小横线。报错信息中会有类似“Error in 文件名 (line 行号)”这样的提示，其中的行号就是代码前的这个行号。小横线说明这一行是可执行代码，而不是注释。鼠标左键小横线，此时横线变为红点，说明在这个位置设置了一个断点。脚本运行时，遇到断点会自动暂停，并进入调试模式。此时可以用 `continue` 继续执行，也可以 `step` 逐行执行，也可以 `step in` 进入调用的子函数执行。初学者一般熟悉 `step` 即可。

进入调试模式时，可以观察 workspace 里的变量，看是否和预想一样，也可以在 command window 里输入其他代码或运行其他脚本/函数。在调试模式中，可以通过左键断点（红点）和小横线来设置新的断点和取消旧的断点。也可以按 `quit debugging` 退出调试模式。

分段是类似断点的一种运行方式，参考上一篇中%%注释的说明。当脚本中有多个%%时，%%相当于分段符号，左键 run and advance 可以实现逐段运行，运行完一段后，相当于进入调试模式，可以在 workspace 里查看运行结果，但其他的操作一般不建议进行。分段运行的另一种用法，是需要输出多个图像的时候，可以运行一段，截图，再运行一段，再截图。

2、常见报错

运行脚本后，会在 command window 里出现红色或者黄色的字，就是报错信息与提示。首先，报错信息会给出定位，哪个文件，第几行。有时会在多个位置给出报错，这说明报错那行的调用关系，例如自定义脚本 A 中 n 行调用了自定义函数 B，而自定义函数 B 的 m 行出错了，则报错信息会定位到 A 的 n 和 B 的 m，看起来好像是两个地方出错，其实是 B 的 m 错了，但实际中也可能是调用 B 的方式不对。这种情况尤其会出现在调用系统自带函数时，此时一般是调用方式不对，所以要关注 A 的 n。

给出报错定位后，就是看具体的报错信息了，matlab 中常见的一些报错信息有：

Undefined function or variable '函数/变量名'.

说明使用变量前没有遵循先定义再使用的原则

Inner matrix dimensions must agree.

数值运算中常见错误，参考数值基本运算那篇的矩阵运算部分

Index exceeds matrix dimensions.

超出索引，例如 $a=[1\ 2\ 3]$ ，此时当用到 $a(4)$ 时

虽然定义了 $a(1:3)$ ，但没有定义 $a(4)$ ，此时却用到了 $a(4)$ ，因此报错

本质上，这也是一种未定义先使用的错误

Subscript indices must either be real positive integers or logicals.

参考矩阵赋值中所说的分块赋值，圆括号中表示位置或者序号，当圆括号中出现非自然数时报错

Expression or statement is incorrect--possibly unbalanced (, {, or [.

有开括号而没有关括号，或者有关括号而没有开括号，一般是计算式太长而打错了

Invalid data type. 或者 Data must be numeric.

或者 Undefined function '函数名' for input arguments of type 'double'.

参考前述的数值运算与符号运算的说明，有些函数只用于数值，有些只用于符号，混用时报错。

1.3 MATLAB 的图形

1、图形窗口

在 MATLAB 下建立一个图形窗口由命令(图 1-2-3)figure 完成，（或命令窗口 FILE-NEW-FIGURE 选项），每执行一次 figure 命令就产生一个图形窗口，可

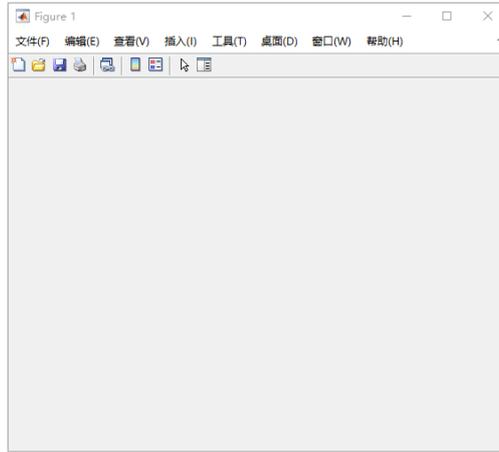


图 1-2-3 MATLAB 图形窗口

以同时产生若干个图形窗口，MATLAB 自动把这些窗口的名字添加序号(No. 1, No. 2, ...)作为区别，同时，这些窗口都被自动分配一个句柄，窗口上有 菜单和工具条，其中包括通用的文件操作命令、编辑命令，对图形的坐标轴、线型等特性进行设置的专用工具，还可以为图形添加标注。用菜单命令操作图形窗口简便、直观，这里不作介绍。

关闭图形窗口由 close 命令来完成，每执行一次 close 命令关闭一个当前的图形窗口，要同时关闭所有窗口，用 close all 来完成。

2. 基本二维图形

绘制二维图形最常用的是 plot 函数，对于不同的输入参数，该函数有不同的形式。

(1) plot(y)

当只有一个参数时，plot 以该参数的值为纵坐标，横坐标从 1 开始自动赋值为向量[1 2 3...]或其转置向量，向量的方向和长度与参数 y 相同。例如，下面的命令：

```
y=[0 1 2 1 0];  
  
plot(y)
```

可以绘制曲线，其横坐标为向量[1 2 3 4 5]

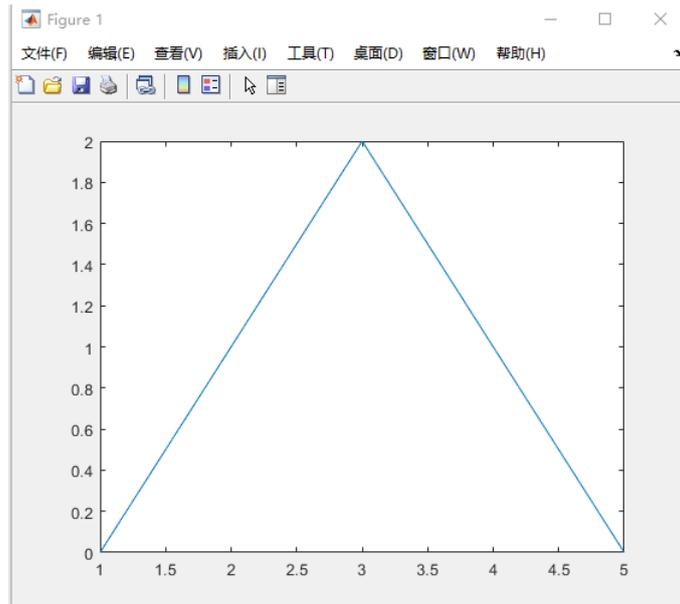


图 1-2-4 plot(y)函数绘制的曲线

(2) plot(x,y)

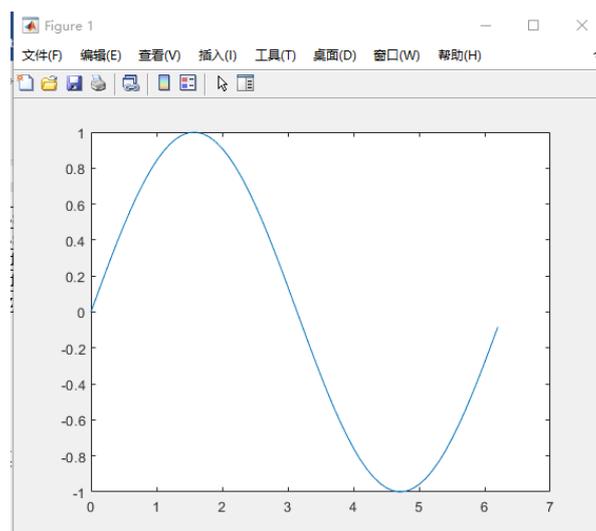
这是最常用的形式。x 为横坐标向量，y 为纵坐标表，例如，下面的命令

```
x=0:0.1:2*pi;
```

```
y=sin(x)
```

```
plot(x,y)
```

绘出一个周期的正弦曲线。



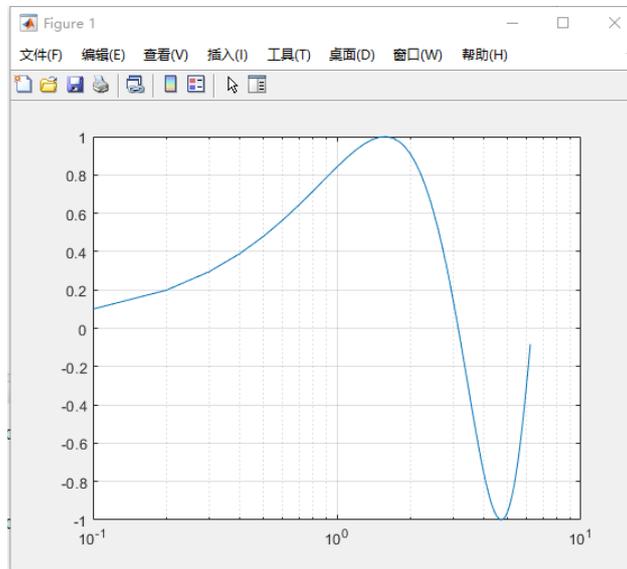
注意：x 和 y 必须方向相同，长度相等，否则 MATLAB 将提示错误。

(3) 对数坐标曲线

三个函数：`semilog`、`semilogx`、`loglog` 可以绘制二维对数坐标曲线，这几个函数的用法和 `plot` 函数相同。例如

```
x=0:0.1:2*pi;  
y=sin(x)  
semilogx(x,y)  
grid on
```

绘制出的正弦横坐标为对数坐标，并用“`grid on`”命令为图形窗口添加了网格。



1.4 M 文件编程

M 文件是 MATLAB 所特有的使用该语言编写的磁盘文件。所有的 M 文件都是以“.m”作为文件的扩展名。建立一个新的 M 文件的方法是在主菜单“File”下选择“NEW”→“M—FILE”，然后会出现 MATLAB 提供的编辑器，MATLAB Editor/Debugger 如图 1-3-1。MATLAB 中的 M 文件可以分为两种类型，一种是脚本，另一种是函数。

脚本

在 MATLAB 中，既不接受输入参数也不返回输出参数的 M 文件称之为脚本。这种 M 文件是在 MATLAB 的工作空间内对资料进行操作的。

当用户在 MATLAB 中调用一个脚本文件，MATLAB 将执行在该脚本文件中所有可识别的命令。脚本文件不仅能够对工作空间内已经存在的变量进行操作，还能够使用这些变量创建新的数据。

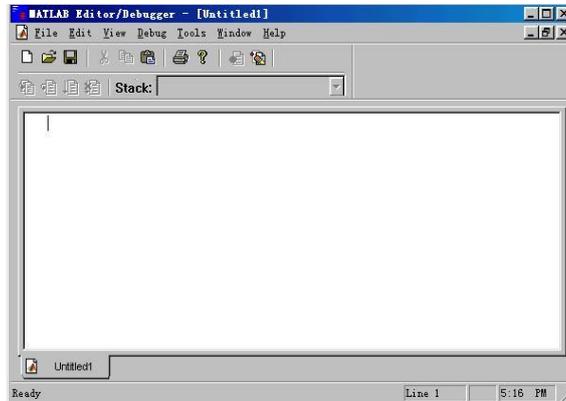


图 1-3-1 M 文件编辑器

尽管脚本文件不能返回输出参数，但其建立的新的变量却能够保存在 MATLAB 的工作空间之中，并且能够在之后的计算中被使用。除此之外，脚本文件还能够使用 MATLAB 的绘图函数来产生图形输出结果。

2. 函数

MATLAB 中的函数是指那些能够接收并输出参数的 M 文件。在 MATLAB 中，函数名和 M 文件名可以是相同的，但函数的对变量的操作是在其函数体内，这种操作与 MATLAB 在工作空间内对变量的操作性质是不同的。

这里，将用一个例子来帮助读者更好的理解这一点。

例如，对于函数“rank”，该函数用来求矩阵的秩，查看该函数的代码。在 MATLAB 的命令窗口中输入

```
>> type rank
```

将显示函数“rank”的代码如下

```
function r = rank(A,tol)

%RANK Matrix rank.

% RANK(A) provides an estimate of the number of linearly
% independent rows or columns of a matrix A.
```

```

%
% RANK(A,TOL) is the number of singular values of A
% that are larger than TOL. By default, TOL = max(size(A)) * eps(norm(A)).
%
% Class support for input A:
%     float: double, single
% Copyright 1984-2015 The MathWorks, Inc.
s = svd(A);
if nargin==1
    tol = max(size(A)) * eps(max(s));
end
r = sum(s > tol);

```

在 MATLAB 中，函数代码的第一行总是以关键词“function”开始，同时，该行也给出了函数名和参数的状态。在本例中，函数代码的第一行给出了函数名“rank”，还显示该函数有两个输入参数及一个输出参数。以“%”号开头的语句在 MATLAB 中被用来表示该行是对该函数的注释，这些注释行将在用户使用帮助命令查询该函数时显示。

函数代码的最后一部分，就是 MATLAB 中实现该函数功能的执行语句。这部分代码实际上才是函数代码的主体。函数体中的所有参数如 A，tol，r 都是函数内部变量，这些变量与 MATLAB 工作空间内的变量是分离的。

§2 Simulink 简介

1、Simulink 简介

Simulink 是 MathWorks 公司开发的又一个产生重大影响的软件产品。它能准确的分析各种复杂的系统模型，在 1990 年，Mathworks 为 MATLAB 提供了新型的控制系统模型输入与仿真工具，并取名为 Simulink，其中包含了两大主要部分：仿真平台和系统仿真模型库。Simulink 可以用来建模、分析和仿真各种动态系统，包括连续、离散以及各种混合系统它提供了鼠标拖放式的方法建立系统框图，通过丰富的功能块，快速建立动态仿真模型。该软件有两个特别明显的功能:仿真与链接。也就是说，可以直接利用鼠标在模型窗口中画出所需要的控制系统模型，然后再利用该软件提供的功能来对控制系统直接进行模拟。很明显，这种做法使得一个原本很复杂的系统变得相当容易输入。Simulink 的出现，使得 Matlab 在控制系统仿真以及电脑辅助设计(CAD)中的应用开创了崭新的一页。

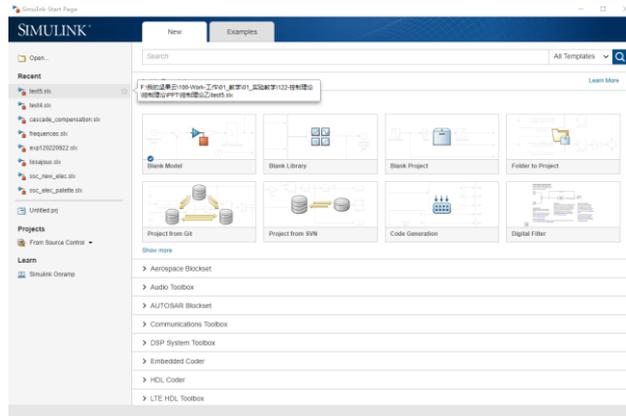
Simulink 是 Matlab 环境下的模拟工具，其文件类型为.mdl, Simulink 为用户提供了方便的图形化功能模块，以便连接一个模拟系统，简化设计流程，减轻设计负担。更重要的是，Simulink 能够用 Matlab 自身的语言或其它语言，根据 s.函数的标准格式，写成自定义的功能模块。因此其扩充性很强，同时也能调用.dll 文件类型的应用程序，实现与其集成应用的目的。在倒立摆系统的仿真过程中会用到 Simulink 的基本操作和用法。

Simulink 是 MATLAB 的重要组成部分，但它具有相对独立的功能和使用方法。Simulink 已经成为在动态系统建模和仿真方面应用最广泛的软件包之一，确切地说，它是对动态系统进行建模、仿真和分析的一个软件包。它支持线性和非线性系统、连续时间系统、离散时间系统、连续和离散混合系统。

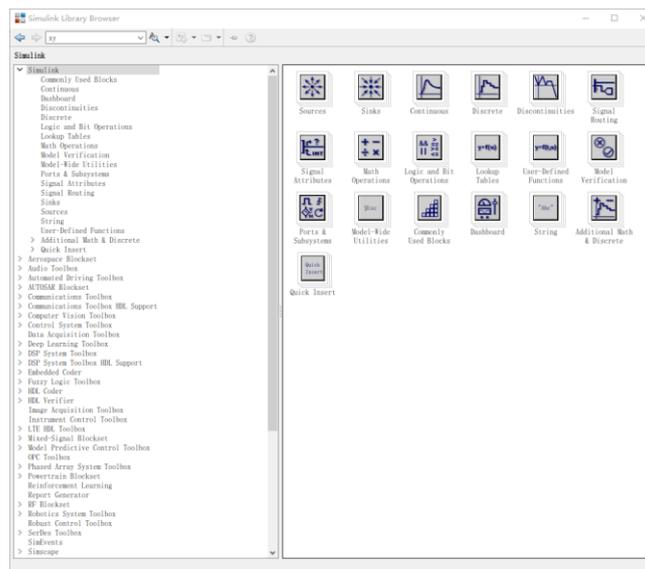
Simulink 基本操作

运行 Simulink 方式有三种

- a、在 MATLAB 的命令窗口直接键入“simulink”
 - b、单击 MATLAB 工具条上的 Simulink 图标
 - c、在 MATLAB 菜单上选择 File—New—Simulink Model
- 选择任一种方式运行后会显示 Simulink Start Page 页面，



双击 Blank Model 即可进入 Simulink academic use 界面，点击 Library Browser(), 进入 Simulink Library Browser 窗口，鼠标左键单击 Simulink 节点，打开模块库窗口，在下拉菜单，同样左键单击模块库，右侧展开库中所包含的各种模块。如下图 Continuous 库包含的各模块：

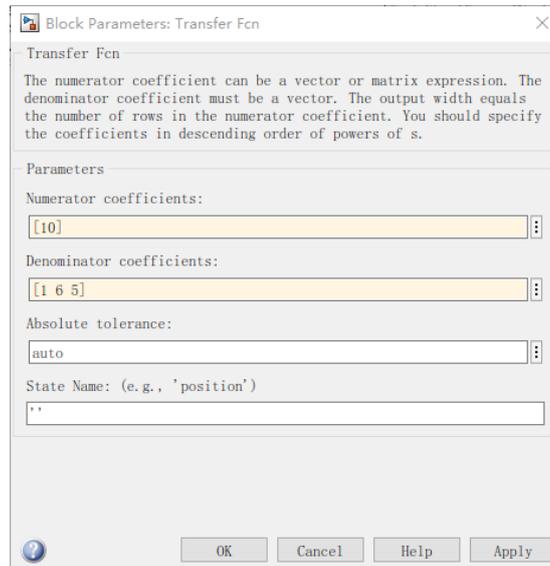


2、Simulink 仿真使用

例：二阶系统的开环传递函数为 $\frac{10}{s^2+6s+5}$ ，在 Simulink 仿真环境中，组成上述闭环系统的仿真框图，观察并记录单位阶跃响应曲线。

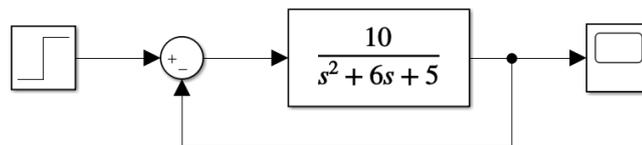
(1)、在 source 库中选择 step 模块，在 Continuous 库中模块 Transfer Fcn 模块，在 Math Operation 库中选择 Sum 模块，在 Sinks 库中选择 scope 模块，分别拖拽到 Simulink academic use 中。

(2)、双击 transfer Fcn 模块，根据传递函数分别修改 Numerator coefficients、Denominator coeffs 参数，如下图所示

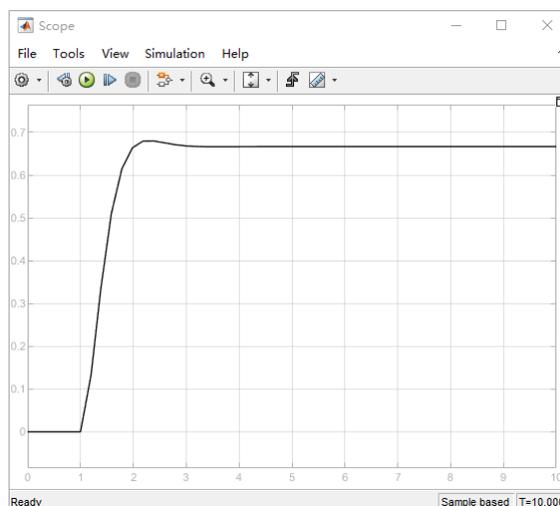


(3)、双击 sum 模块，将++改为+-。

(4)、如下图所示，连接各模块，构成闭环系统。



(5)、点击工具栏上的 Run 按钮，开始仿真，双击 scope，即可看到输出响应波形，结果如下图：



3. Matlab 控制系统常用函数命令使用介绍

1)、G=tf(num,den)

功能：建立实部或复数传递函数模型或将状态方程、或零级增益模型转化成传递函数形式。

说明：num 为分子，den 为分母，G 为系统传递函数。

例 1：建立 M 文件，英文格式命名并保存，输入

```
num = 1;           % 分子多项式的系数
den = [2, 3];     % 分母多项式的系数
G = tf(num, den)  % 求传递函数
```

保存后，运行输出为：

```
G =
    1
-----
 2 s + 3
```

2)、G=zpk (Z, P, K)

功能：建立零极点传递函数模型。

说明：Z 为零点、P 为极点、K 为增益。

例 2：由零极点得出系统传递函数模型。

```
num = 1;           % 分子多项式的系数
den = [1,3,2];     % 分母多项式的系数
Z = roots(num)
P = roots(den)
G = zpk(Z,P,1)
```

传递函数为

```
G =
    1
```

(s+2) (s+1)

Continuous-time zero/pole/gain model.

3)、 $G=ss(A,B,C,D)$

功能：创建实数或复数的状态空间模型，或者将传递函数和零极点增益模型转为状态空间模型。

说明：A，B，C，D 分别为状态方程的系数矩阵。

4)、 $[num,den] = ss2tf(A,B,C,D,iu)$

功能：变系统状态空间形式转换为传递函数模式

说明：系统的状态空间方程表示为

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

相应的传递函数为 $H(s) = \frac{num(s)}{den(s)} = C(sI - A)^{-1}B + D$

$[num,den] = ss2tf(A,B,C,D,iu)$ 可将状态空间表示转换为对应的传递函数表示，iu 用于指定的变换所需要的输入量。

5)、 $[z,p,k]=ss2zp(A,B,C,D,iu)$

功能：将状态空间表示转换为零极点增益形式表示

6)、 $[A,B,C,D]=tf2ss(num,den)$

功能：将传递函数形式表示转换为状态空间形式表示

7)、 $[z,p,k]=tf2zp(num,den)$

功能：将系统传递函数模式表示转换为零极点增益形式表示

8)、 $[A,B,C,D]=zp2ss(z,p,k)$

功能：将系统零极点增益形式转换为状态空间形式。

9)、 $[num,den]=zp2tf(z,p,k)$

功能：将零极点增益形式转换为传递函数模式。

例 3、将系统 $H(s) = \frac{2s+5}{s^2+4s+4}$ ，变换成状态空间形式。

```
num = [0 2 5;1 4 4];  
den = [1 0.2 1];  
[A,B,C,D]=tf2ss(num,den)
```

输出为:

```
A =  
    -0.2000    -1.0000  
    1.0000         0  
B =  
     1  
     0  
C =  
     2.0000     5.0000  
     3.8000     3.0000  
D =  
     0  
     1
```

例 5、找出系统 $H(s) = \frac{s^2+0.5s+1}{s^2-2s+1}$ 的零极点与增益。

```
num = [1 0.5 1];  
den = [1 -2 1];  
[z,p,k]=tf2zp(num,den)
```

输出为

```
z =  
    -0.2500 + 0.9682i  
    -0.2500 - 0.9682i  
p =
```

```

1
1
k =
1

```

10)、`[r,p,k]=residue(num,den)`

功能：将求出两个多项式 $B(s)$ 和 $A(s)$ 之比的部分分式展开留数、极点和直接项。

说明：
$$\frac{B(s)}{A(s)} = \frac{r(1)}{s-p(1)} + \frac{r(2)}{s-p(2)} + \dots + \frac{r(n)}{s-p(n)} + k(s)$$

例 6、求出传递函数 $\frac{B(s)}{A(s)} = \frac{2s^3+5s^2+3s+6}{s^3+6s^2+11s+6}$ 的部分分式表达式

```

num = [2 5 3 6];
den = [1 6 11 6];
[r,p,k]=residue(num,den)

```

输出为：

```

r =
-6.0000
-4.0000
3.0000
p =
-3.0000
-2.0000
-1.0000
k =
2

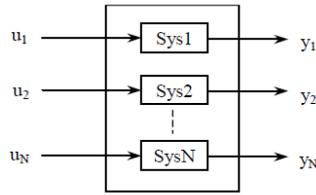
```

表达式为：

$$\frac{B(s)}{A(s)} = \frac{2s^3 + 5s^2 + 3s + 6}{s^3 + 6s^2 + 11s + 6} = \frac{-6}{s+3} + \frac{-4}{s+2} + \frac{3}{s+1} + 2$$

11)、`sys=append(sys1,sys2)`

功能：将多个 LTI 系统组合（将输入与输出组加起来）



例 7、

```
G1=tf(1,[1 2 1]);
G2 = tf(1,[1 1]);
G=append(G1,G2)
```

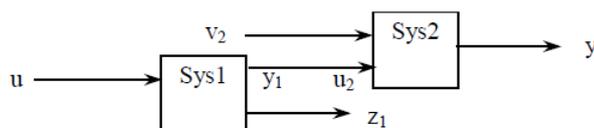
输出为：

```
G =
From input 1 to output...
      1
1: -----
   s^2 + 2 s + 1
2: 0
From input 2 to output...
1: 0
      1
2: ----
   s + 1
Continuous-time transfer function.
```

12)、G=series(G1,G2)

功能：实现 $G1 * G2$ ，两个子系统必须连续时间系统或者具有相同采样周期的离散时间系统。

扩展：G=series(G1,G2,outputs1,inputs2) outputs1 和 inputs2 用于指定 sys1 的部分输出与 sys2 的部分输入进行连接，如下图所示



例 8、使用 series 将 $G1 = \frac{1}{s+2}$ 于 $G2 = \frac{1}{2s+3}$ 串联

```
G1=tf(1,[1 2]);
```

```
G2 = tf(1,[2 3]);
```

```
G=series(G1,G2)
```

输出为:

```
G =
```

```
1
```

```
-----
```

```
2 s^2 + 7 s + 6
```

13)、 $G=\text{parallel}(G1,G2)$

功能: 实现 $G1+G2$

14)、 $G=\text{feedback}(G1,G2,\text{sign})$

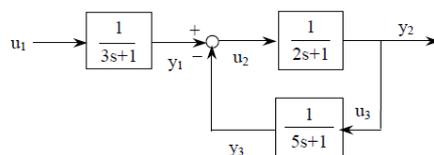
功能: 实现 $G1, G2$ 构成的反馈, 其中 $G1$ 为前向通道, $G2$ 为反馈通道, $\text{sign}=1$ 为正反馈, $\text{sign}=-1$ 为负反馈。

15)、 $G=\text{connect}(G1,Q,\text{inputs},\text{outputs})$

功能: 实现框图连接, 输出整个系统的传递函数。

说明: $G1$ 由 `append` 生成的无连接对角方块系统, Q 矩阵用于指定系统 sys 的内部连接关系, 其中矩阵的每一行对应一个输入, 其第一个元素为输入编号, 其后为连接该输入的输入编号, 如采用负连接, 则以负值表示。 `inputs` 和 `outputs` 用于指定无连接系统中的某些输入/输出保留作为外部的输入输出

例 9、根据系统框图, 求出系统总的传递函数



```
G1 = tf(1,[3 1]);
```

```
G2 = tf(1,[2 1]);
```

```
G3 = tf(1,[5 1]);
```

```
G = append(G1,G2,G3);
```

```
Q = [2 -3 1;3 2 0];
```

```

inputs = 1;

outputs = 2;

GC = tf(connect(G,Q,inputs,outputs))

```

输出为:

```

GC =
      0.1667 s + 0.03333
-----
s^3 + 1.033 s^2 + 0.4333 s + 0.06667

```

即为: $GC = \frac{5s+1}{30s^2+31s^2+13s+2}$

initial

语法: `initial(G, x0, Ts)`

`[y, t, x] = initial(G, x0, Ts)`

功能: 求连续系统的零输入响应。

说明: G 必须为系统的状态空间模型, x₀为初始条件, T_s为时间点, 可省略, y为输出响应, t为时间向量, 可省略, x为状态变量响应, 可省略。

impulse

语法: `impulse(G, Ts)`

`[y, t, x] = impulse(G, Ts)`

功能: 连续系统的单位冲激响应。

说明: G为系统模型, 可为传递函数、状态方程、零极点增益形式; y为时间响应; t为时间向量; x为状态变量响应; Ts为时间点, t, x, Ts可省略。

step

语法: `step(G, Ts)`

`[y, t, x] = step(G, Ts)`

说明: 参数说明见 impulse。

lsim

语法: `lsim(G,U,Ts)`

`Lsim(G1,G2,...,U,Ts)`

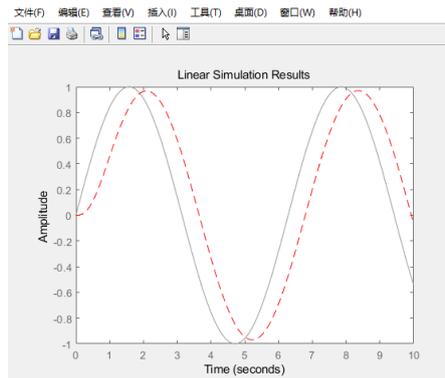
`[y,t,x] = lsim(G,U,Ts)`

说明: U 为输入序列, 每一列对应一个输入; Ts 为时间点, U 的行数与 Ts 相对应; t 和 x 可省略。

例 10: 传递函数为 $G = \frac{10}{s^2+5s+10}$, 在任意输入下的响应, 输入信号为正弦波。

```
t=0:0.1:10;  
u = sin(t);  
G = tf(10,[1 5 10]);  
lsim(G,'r--',u,t)
```

输入为:



tzzero

语法: `z=tzzero(G)`

`[z,gain]=tzzero(G)`

说明: 计算系统的零点和增益。

damp

语法: `damp(sys)`

`[wn,zeta] = damp(sys)`

`[wn,zeta,P] = damp(sys)`

功能：计算固有频率和阻尼比。

说明：其中，wn 为固有频率 ω_n ；zeta 为阻尼系数 ζ 。

例 11：计算系统 $G = \frac{10}{s^2+5s+6}$ 的极点，阻尼系统，固有频率。

程序清单：

```
num = [10];  
den = [1 5 6];  
G = tf(num,den);  
[wn,zeta] = damp(G)
```

输出内容：

```
wn =  
    2.0000  
    3.0000  
zeta =  
    1  
    1
```

例 12：计算 $G = \frac{10}{s^2+5s+6}$ 的频域特性。

程序清单：

```
num = [10];  
den = [1 5 6];  
w = 1;  
Gw = polyval(num ,j*w)./polyval(den,j*w)  
Aw = abs(Gw)  
Fw = angle(Gw)
```

输出：

```
Gw =  
1.0000 - 1.0000i
```

```
Aw =  
    1.4142  
Fw =  
   -0.7854
```

bode

函数: `bode(sys)` %绘制 bode 图
`[mag, pha] = bode(sys, w)` %绘制 w 对应的幅值和相角
`[mag, pha, w] = bode(sys)` %绘制幅值、相角、频率

说明: `sys` 为系统模型, `w` 为频率, `mag` 为幅值, `pha` 为相角。

nyquist

函数: `Nyquist(sys)` %绘制 nyquist 曲线
`[Re, Im] = nyquist(sys, w)` %由 w 得出对应的实部和虚部
`[Re, Im, w] = Nyquist(sys)` %得出实部、虚部、频率。

说明: `sys` 为系统模型, `w` 为系统频率, `Re` 为频率特性的实部, `Im` 为频率特性的虚部。

margin

函数: `margin(sys)` %绘制 bode 图, 并标出幅值裕量和相位裕量
`[Gm, Pm, Wcg, Wcp] = margin(sys)` %得出幅值裕量与相位裕量

说明: `Gm` 为幅值裕量, `Pm` 为相位裕量, `Wcg` 为幅值裕量对应的频率, `Wcp` 为相位裕量对应的频率, 若 `Wcg, Wcp` 的值为 `nan` 或 `inf`, 则对应的幅值裕量和相位裕量为无穷大。

rlocus

函数: `rlocus(sys)` %绘制根轨迹
`[r, k] = rlocus(sys)` %计算闭环极点和对应的 k
`r = rlocus(sys, k)` %k 为增益, 返回该增益下的根轨迹各个点构成的复数矩阵。

pzmap

函数: `[p, z] = pzmap(sys)` %绘制系统的零极点图。

4 Matlab 常用绘图命令

命令	说明
grid on	在所画出的图形坐标中加入栅格
grid off	在所画出的图形坐标中移除栅格
hold on	把当前图形保持在屏幕上不变，同时允许在这个坐标内绘制另外一个图形。
hold off	新图覆盖旧图
axis([xmin xmax ymin ymax])	设定坐标轴范围
text(x,y,'字符串')	在图形的指定坐标位置 (x, y) 处，标示单引号括起来的字符串。
gtext('字符串')	利用鼠标在图形的某一位置标示字符串
title('字符串')	在所画图形的最上端显示说明该图形标题的字符串。
xlabel('string'),ylabel('string')	设置 x, y 坐标轴的名称，输入特殊的文字需要用反斜杠 (\) 开头。
legend('string1','string2',...,'stringn')	在屏幕上开启一个小视窗，然后依据绘图命令的先后次序，用对应的字符串区分图形上的线。
subplot(m,n,k)	分割图形显示窗口 m: 上下分割个数, n: 左右分割个数, k: 子图编号
semilogx	绘制以 x 轴为对数坐标 (以 10 为底), y 轴为线性坐标的半对数坐标图形
semilogy	绘制以 y 轴为对数坐标 (以 10 为底), x 轴为线性坐标的半对数坐标图形

§3 SimCap 模型库概述

S Simscape 是在 Simulink 环境下创建物理系统模型的工具和语言,Simscape 库由基础库与高级库构成：基础库涵盖电、磁、机械、热、液、气、两相流等物理域，如图 1 所示；高级库涵盖电气、传动、多体、流体等物理域，如图 2 所示；此外支持自定义模块，sscnewfile(‘自定义模块名称’)。

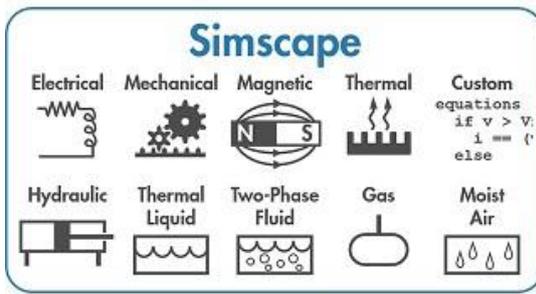


图 1-1-1 基础库



图 1-1-2 高级库

基础库中模块主要分为 Elements(元件)、Sensors(传感器)、Sources(信号源)。元件例如电阻电容、阻尼弹簧质量块、每个物理域的参考、物理域之间的信号转换模块等，传感器例如电压表电流表、位移速度传感器等，信号源例如电压源电流源、力源扭矩源等。

此外基础库中还包括物理信号模块、用于物理信号与 Simulink 信号相互转换的模块、求解器模块等。Simulink 中的 Scope、PID 等模块亦可以用于 Simscape 的参数显示与控制。

其中，simulink 电路仿真常用到的信号转换模块为：PS-Simulink Converter 为物理信号-simulink 信号转换器，求解器（solver configuration）来启动仿真。

SimCap 电路仿真举例：

在 Matlab R2019a 的命令行窗口输入：simulink，回车，即可打开 simulink 界面，在图 1-1-3 中，找到 simscape

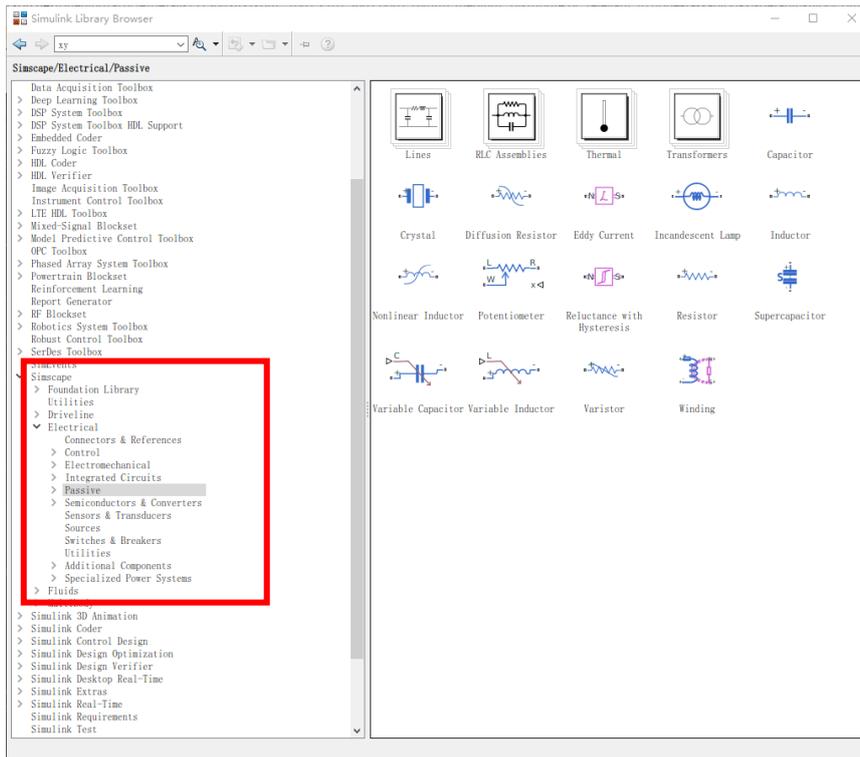


图 1-1-3 simcape

在 simcape/Foundation Library/Electrica 目录下找到需要的电子元件，搭建实验所需要的电路图，以惯性环节为例，搭建好的电路图如图 1-1-4 所示，输出如图 1-1-5 所示

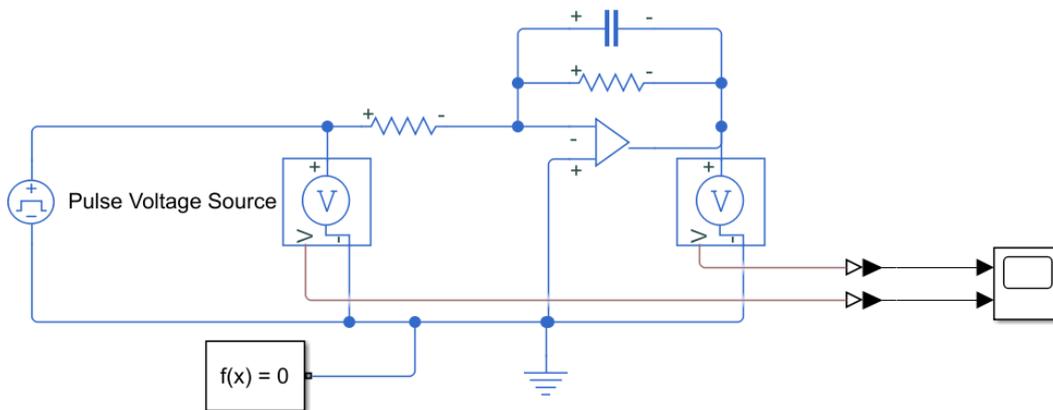


图 1-1-4 惯性环节电路图

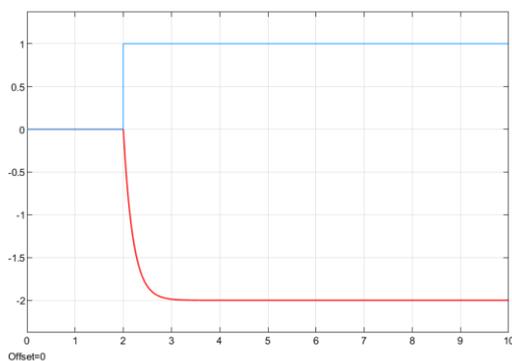


图 1-1-5 惯性环节输出

图 1-1-4 中器件的位置分别如下：

1、基本电阻、电容、放大器的位置为:Simcape/Foundation Library/Electrical/Electrial Elements。

2、PS-Sinulink Converter (—▶▶) 及求解器 ($f(x) = 0$) 的位置为:Simcape/Utilities, 如图 1-1-6 所示

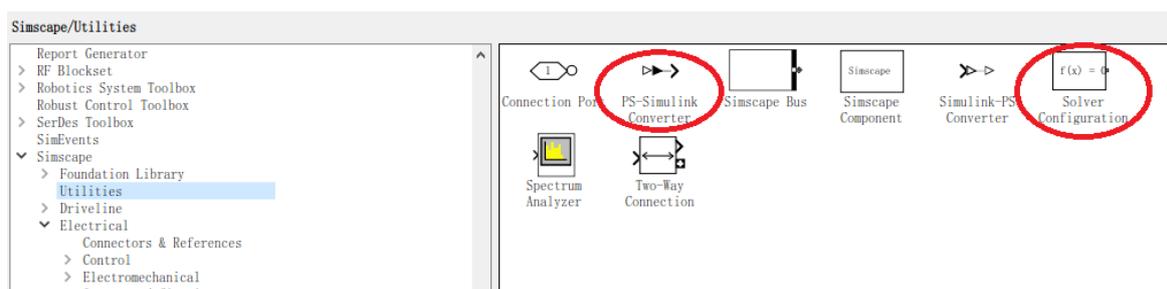


图 1-1-6 Ps-Simulink 及 Solver Configuration 位置

3、电压传感器位置为: Simcape/Foundation Library/Electrical/Electrial Sensor。

4、示波器输入接口设置，见图 1-1-7

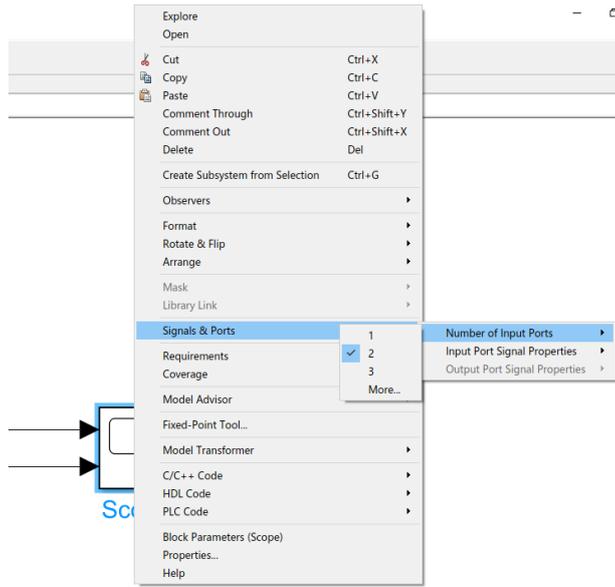


图 1-1-7 示波器输入端口设置

5、双击电路图中 R 或 C，即可就修改对应器件的阻、容值。

图 1-1-5 示波器输出白底图片设置方法：

1、选择 View->style，即可出现图 1-1-8

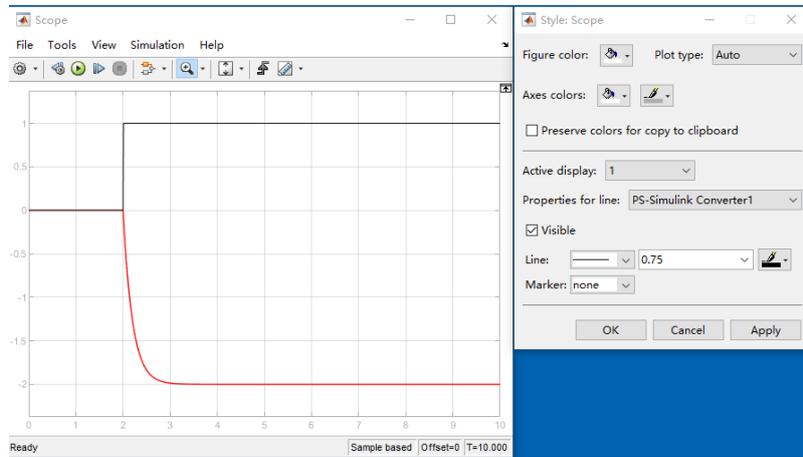
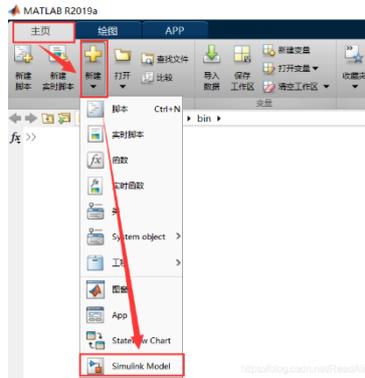


图 1-1-8 图片设置

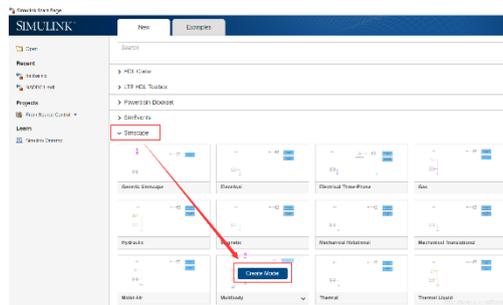
2、在 Figure color 与 Axes colors 下拉菜单中选择白色，Properties for lines 选择需要修改曲线的颜色。

用 Simscape 实现三维物理仿真-倒立摆系统举例

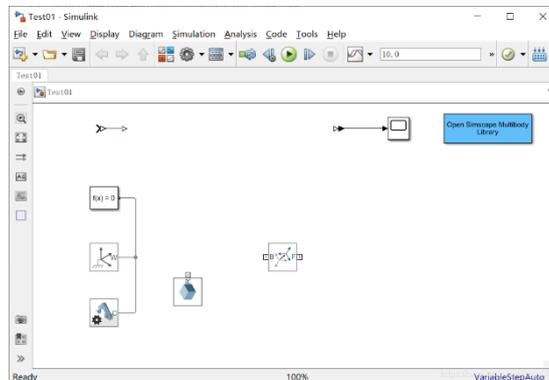
1 打开 Simulink,主页->新建->Simulink Model。



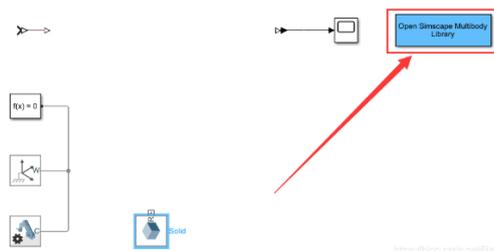
2 创建一个 Simscape Multibody Model。



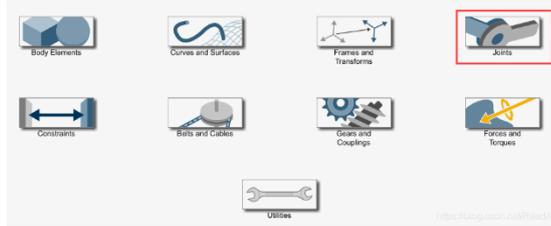
3 保存



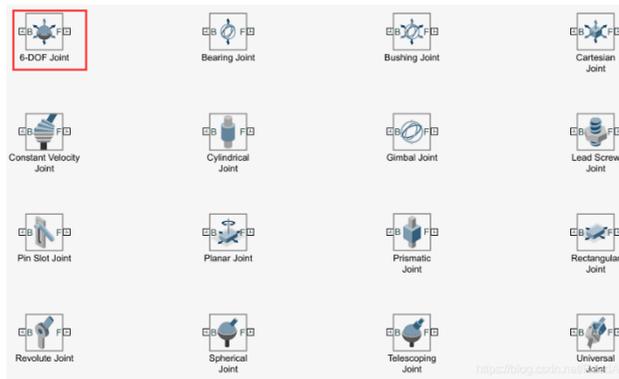
4 添加运动副/关节 (Joint)。鼠标左键双击打开 Open Simscape Multibody Library



6、找到“Joints”并点开



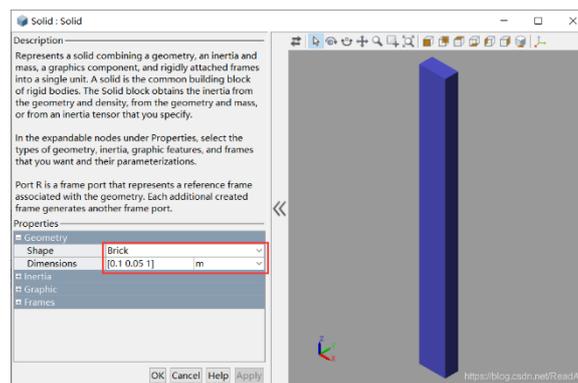
选择第一个“6 DOF Joint”，按鼠标左键不动将其拖出到仿真面板上



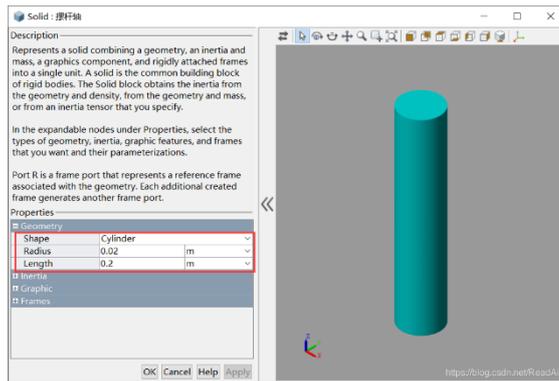
7、创建四个 Solid 模块



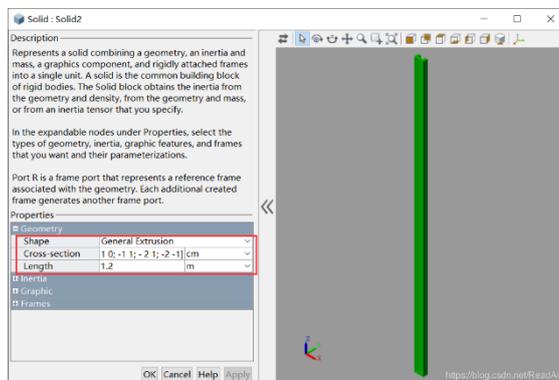
摆杆：Shape : Brick, Dimensions : [0.1 0.05 1] m



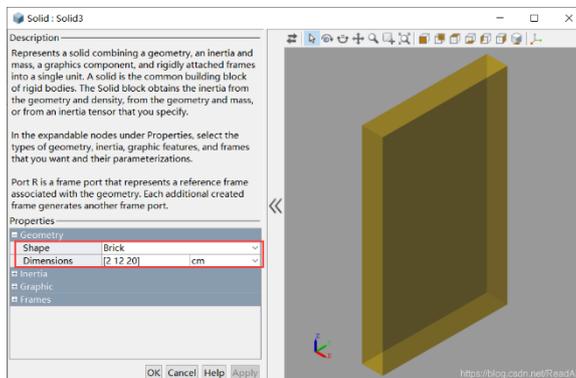
摆杆轴：Shape : Cylinder; Radius : 0.02 m; Length : 0.2 m



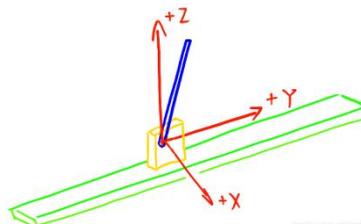
滑块导轨: Shape : General Extrusion; Cross-section : [2 -1;2 1;1 1;1 0;-1 0;-1 1;- 2 1;-2 -1] cm; Length : 1.2 m



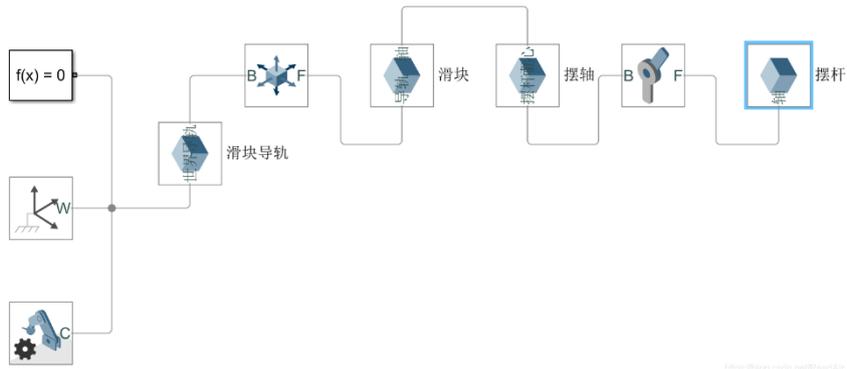
滑块: Shape : Brick; Dimensions : [2 12 20] cm



7、建立合适的坐标系,单摆系统世界坐标系如下图,因此按照此坐标系修改各组件之间的参考坐标,

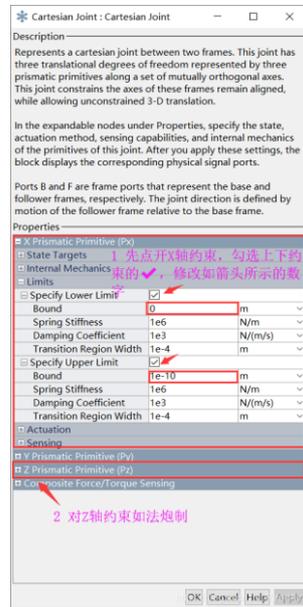


连接好的图形是如下图所示:

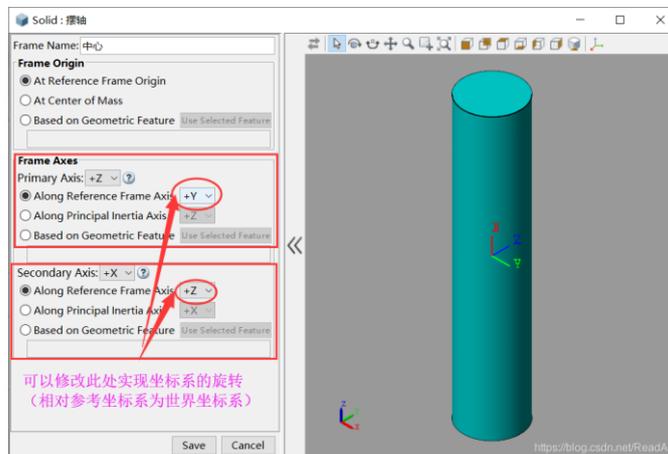


其中需要注意的几点：

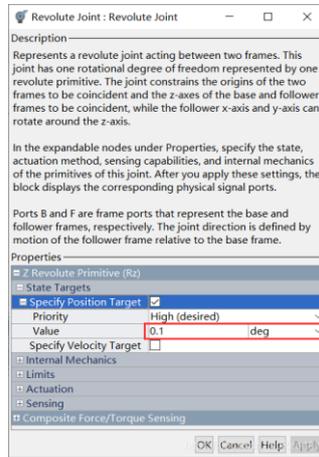
1) 滑块与导轨之间是一个单自由度滑动连接，根据世界坐标系，可滑动轴在 Y 轴方向，我们选择的三自由度滑动模块  必须限制 X 和 Z 轴。具体修改方式如下



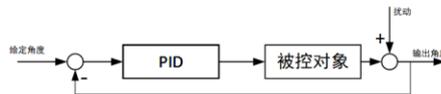
2) 当某些 solid 机构坐标系与关节限定的轴不一致时（比如单轴转动副，转轴必须为 Z 轴），可用通过修改 Solid 机构的坐标系实现坐标系旋转变换。



3) 为了运行仿真开始时单摆可以摆动，需要使单摆初始重心位置不在稳态处，通过修改转动副的初始角度即可。

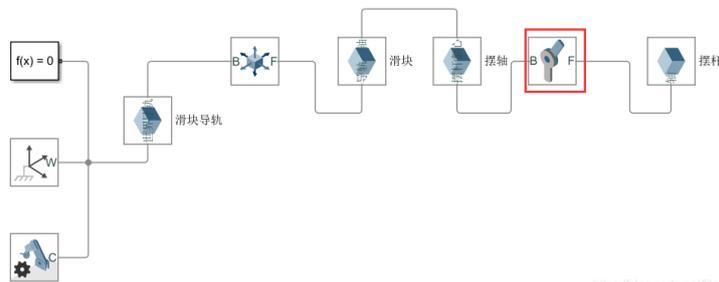


根据闭环系统的控制框图添加传感器。



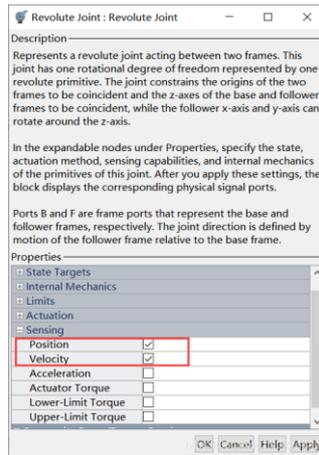
8、为系统添加传感器

(1) 双击编辑 旋转关节



<https://blog.csdn.net/FeedAir>

(2) 在属性中找到传感器 (Sensing), 在其子栏目中勾选位置 (即角度) 和速度 (即角速度), 然后 Apply



(3) 可以看到这个模块多了两个输出 q (角度)、 w (角速度):



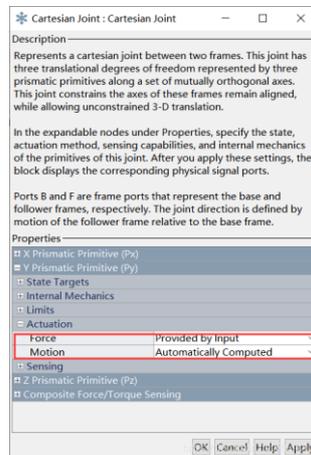
9、添加控制器

控制器实质是控制滑块 Y 轴向的受力（Y 轴是滑块的自由轴）

(1) 双击编辑 三轴平动关节



(2) 在属性中找到驱动（Actuation），将受力(Force)选为“由输入提供（Provided by input）”，运动（Motion）选为“自动计算（Automatically computed）”，然后点击 Apply

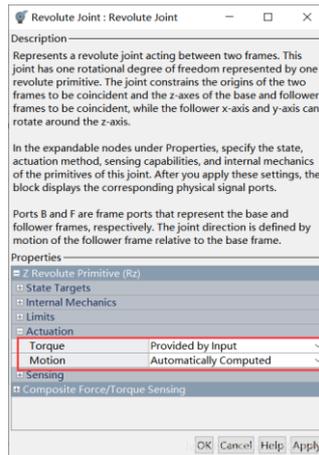


(3) 可以看到模块多了一个 Y 轴受力输入:

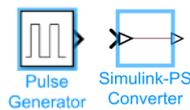


10 添加噪声

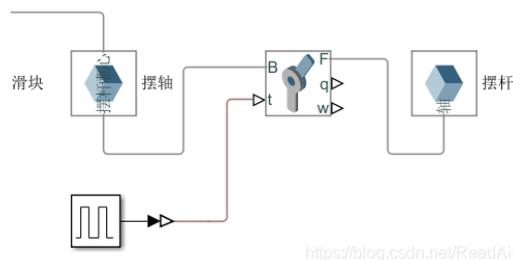
(1) 在之前的旋转关节中找到驱动，添加一个扭矩（Torque）输入:



(2) 从 Simulink 库拖出一个脉冲发生器和一个信号转换器：脉冲发生器 (Simulink -> Sources)和信号转换器 (Simscape -> Utilities):



(3) 如图连接接入旋转关节：



(4) 双击设置脉冲发生器参数：脉冲幅度：5;周期：5;脉冲宽度：5%;脉冲延时：1



11、添加 PID 控制律

仿真是一个完全理想的环境,不存在机械死区和摩擦力 (尽管你可以自己添加), 因此积

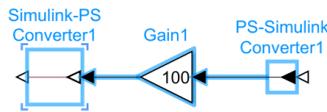
分部分我们可以不用，所以我要展示的严格来说是个 PD 控制器。

其次，由于误差等于输入（给定）减去输出，而输入恒为零，因此误差就是负的输入。

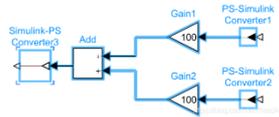
此外，误差的微分即角速度，我们这里可以用传感器直接读出。即可省去微分的计算过程。

了解这些后，我们开始着手于搭建控制器：

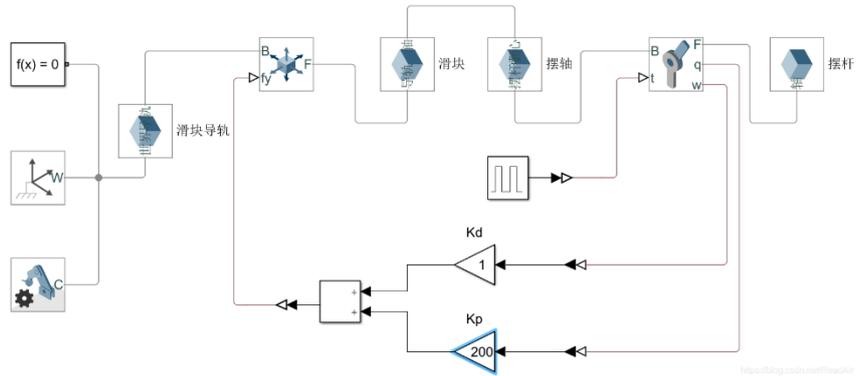
(1) 从 Simulink 库中拖出两个增益模块（Gain），以及对应的两种信号转换器，并按图示连接：



(2) 添加一个加模块（Add），按图示连接：



(3) 现在按照我们的控制框图完成全部连接，取 $K_p=200, K_d=1$ ，如图所示连接：



§4 python-control 使用介绍

Python 库控制系统(`python-control`)是一个 Python 包，能基本实现反馈控制系统的分析和设计。

特征：

- 1、线性输入/输出系统状态方程和频域
- 2、非线性输入/输出系统建模、仿真和分析
- 3、框图代数:串行、并行和反馈联系
- 4、时域响应：零状态,阶跃,冲激
- 5、频域响应：伯德图、奈奎斯特图
- 6、控制系统分析:稳定性、能控性、能观性、稳定裕量
- 7、控制系统设计:特征值配置，LQR，H2, H_∞
- 8、估计设计:线性二次估计量(卡尔曼滤波)等。

`python-control` 包是一组 python 类和函数，用于实现反馈控制系统分析和设计的常见操作。最初的目标是实现完成 Astrom 和 Murray 教科书中反馈系统的示例所需的功能。模块中的很多函数与 MATLAB 控制系统工具箱中可用的命令相兼容。

值得注意的是：`python` 控制包使用 NumPy 和 SciPy。对于 `python-control` 包，向量中必须包含逗号，所以 `[1 2 3]` 一定是 `[1,2,3]`。返回多个参数的函数使用元组。集合使用元组代替大括号。

安装

`python-control` 包可以使用 `pip`，`conda` 或标准的 `distutils/setuptools` 机制进行安装。该包需要 `numpy` 和 `scipy`，绘图例程需要 `matplotlib`。此外，某些例程需要 `slycot` 库才能实现更高级的功能（如 MIMO 功能）。

1、pip 安装

```
pip install slycot # optional
pip install control
```

2、conda 安装

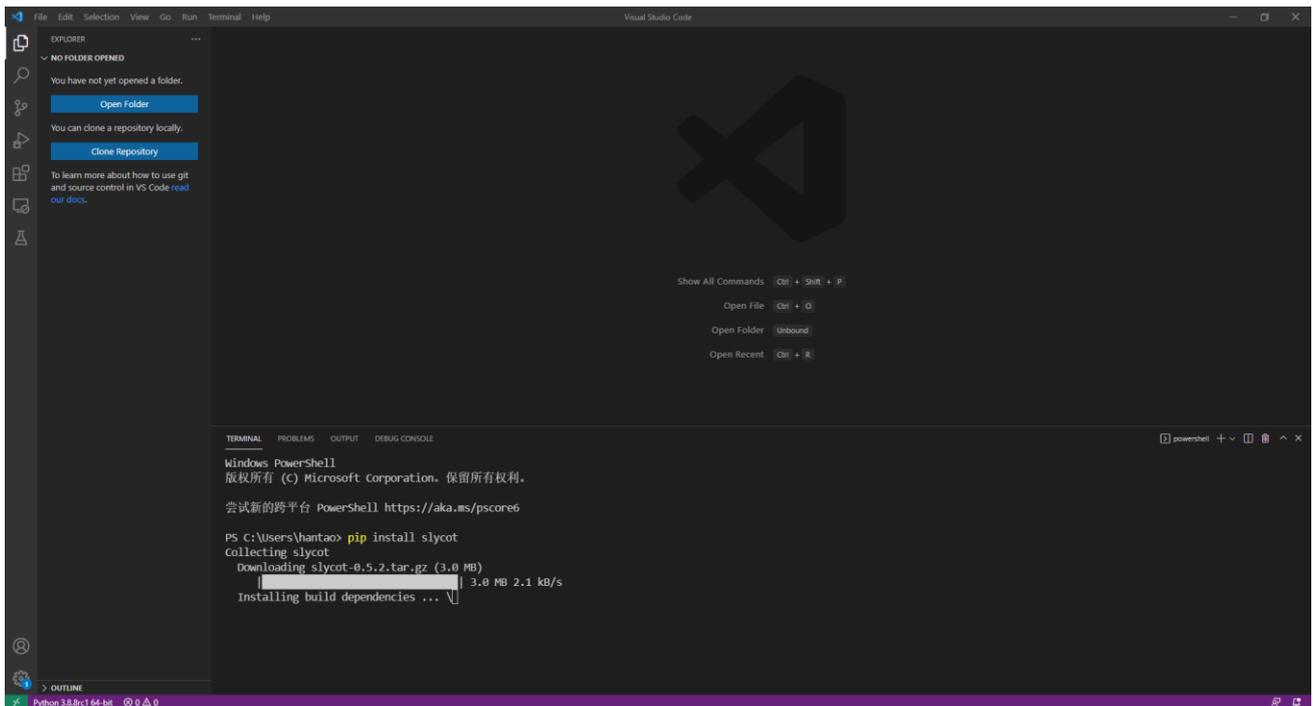
```
conda install -c conda-forge control
```

```
conda install -c conda-forge slycot
```

3、源码安装

下载源码后，解压缩，进入目录，输入命令

```
python setup.py install --user
```



Python 基础

注释：python 中注释使用“#”和代码区分，多行或块注释，使用三个单引号“'''”

```
'''  
  
注释 1  
  
注释 2  
  
'''
```

字符串：字符串是 Python 中最常用的数据类型。我们可以使用引号（'或"）来创建字符串。

变量类型

变量是存储在内存中的值，这就意味着在创建变量时会在内存中开辟一个空间。基于变量的数据类型，解释器会分配指定内存，并决定什么数据可以被存储在内存中。因此，变量可以指定不同的数据类型，这些变量可以存储整数，小数或字符。

变量赋值

Python 中的变量赋值不需要类型声明。每个变量在内存中创建，都包括变量的标识，名称和数据这些信息。每个变量在使用前都必须赋值，变量赋值以后该变量才会被创建。等号 = 用来给变量赋值。等号 = 运算符左边是一个变量名，等号 = 运算符右边是存储在变量中的值。

多个变量赋值

Python 允许你同时为多个变量赋值。例如：

```
a = b = c = 1
```

以上实例，创建一个整型对象，值为 1，三个变量被分配到相同的内存空间上。也可以为多个对象指定多个变量。例如：

```
a, b, c = 1, 2, "john"
```

以上实例，两个整型对象 1 和 2 分别分配给变量 a 和 b，字符串对象 "john" 分配给变量 c。

标准数据类型

在内存中存储的数据可以有多种类型。例如，一个人的年龄可以用数字来存储，他的名字可以用字符来存储。Python 定义了一些标准类型，用于存储各种类型的数据。

Python 有五个标准的数据类型：

- 1、Numbers（数字）
- 2、String（字符串）
- 3、List（列表）
- 4、Tuple（元组）

5、Dictionary (字典)

数字: 数字数据类型用于存储数值。是不可改变的数据类型，这意味着改变数字数据类型会分配一个新的对象。

当指定一个值时，Number 对象就会被创建：

```
var1 = 1  
  
var2 = 10
```

也可以使用 del 语句删除一些对象的引用。

del 语句的语法是：

```
del var1[,var2[,var3[....,varN]]]
```

可以通过使用 del 语句删除单个或多个对象的引用。例如：

```
del var  
  
del var_a, var_b
```

Python 支持四种不同的数字类型：int (有符号整型)、long (长整型，也可以代表八进制和十六进制)、float (浮点型)、complex (复数)

如：int:10; long:51923461L; float:15.20; complex:3.14j

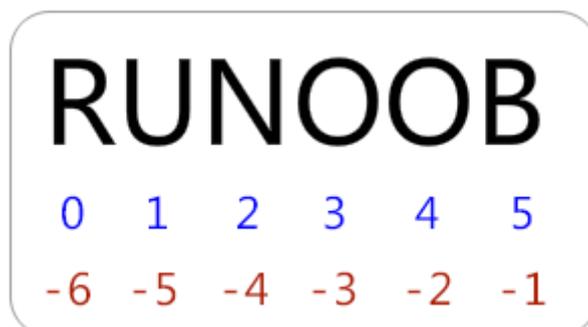
字符串: 字符串或串(String)是由数字、字母、下划线组成的一串字符。一般记为：

```
s = "a1a2· · · an" # n>=0
```

是编程语言中表示文本的数据类型。

python 的字符串列表有 2 种取值顺序：

- 从左到右索引默认 0 开始的，最大范围是字符串长度少 1
- 从右到左索引默认-1 开始的，最大范围是字符串开头



如果你要实现从字符串中获取一段子字符串的话，可以使用 [头下标:尾下标] 来截取相应

的字符串，其中下标是从 0 开始算起，可以是正数或负数，下标可以为空表示取到头或尾。

[头下标:尾下标] 获取的子字符串包含头下标的字符，但不包含尾下标的字符。比如：

```
>>> s = 'abcdef'

>>> s[1:5]

'bcde'
```

当使用以冒号分隔的字符串，python 返回一个新的对象，结果包含了以这对偏移标识的连续的内容，左边的开始是包含了下边界。上面的结果包含了 **s[1]** 的值 **b**，而取到的最大范围不包括**尾下标**，就是 **s[5]** 的值 **f**。

从后面索引：	-6	-5	-4	-3	-2	-1	
从前面索引：	0	1	2	3	4	5	
	+---	+---	+---	+---	+---	+---	
	a	b	c	d	e	f	
	+---	+---	+---	+---	+---	+---	
从前面截取：	:	1	2	3	4	5	:
从后面截取：	:	-5	-4	-3	-2	-1	:

加号 (+) 是字符串连接运算符，星号 (*) 是重复操作。如下实例：

```
str = 'Hello World!'

print str          # 输出完整字符串

print str[0]       # 输出字符串中的第一个字符

print str[2:5]     # 输出字符串中第三个至第六个之间的字符串

print str[2:]      # 输出从第三个字符开始的字符串

print str * 2      # 输出字符串两次

print str + "TEST" # 输出连接的字符串
```

输出为：

```
Hello World!

H

llo

llo World!
```

```
Hello World!Hello World!
```

```
Hello World!TEST
```

Python 列表截取可以接收第三个参数，参数作用是截取的步长，以下实例在索引 1 到索引 4 的位置并设置为步长为 2（间隔一个位置）来截取字符串：

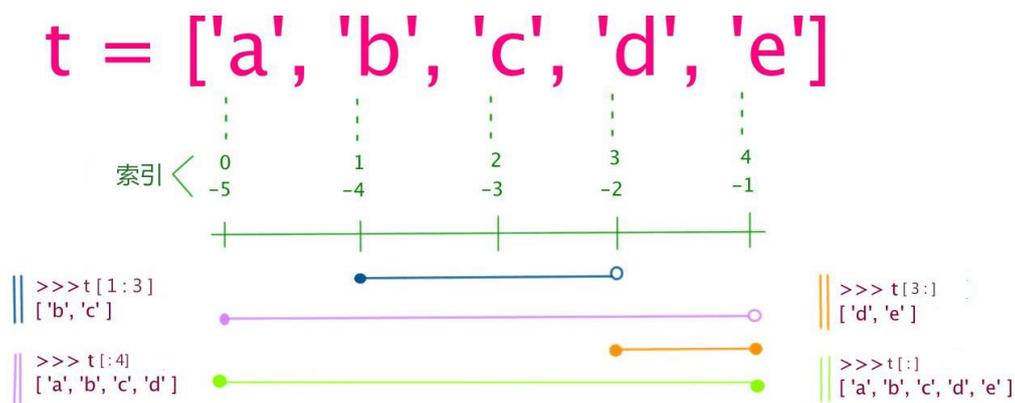
```
      0   1   2   3   4   5   6
>>> letters = ['c', 'h', 'e', 'c', 'k', 'i', 'o']

                2
                ~~~~~

>>> letters[1:4:2]
['h', 'c']
```

列表：List（列表）是 Python 中使用最频繁的数据类型。列表可以完成大多数集合类的数据结构实现。它支持字符，数字，字符串甚至可以包含列表（即嵌套）。

列表用 [] 标识，是 python 最通用的复合数据类型。列表中值的切割也可以用到变量 [头下标:尾下标]，就可以截取相应的列表，从左到右索引默认 0 开始，从右到左索引默认 -1 开始，下标可以为空表示取到头或尾。



加号 + 是列表连接运算符，星号 * 是重复操作。如下实例：

```
#!/usr/bin/python
```

```
# -*- coding: UTF-8 -*-
```

```

list = [ 'runoob', 786 , 2.23, 'john', 70.2 ]

tinylist = [123, 'john']

print list          # 输出完整列表
print list[0]      # 输出列表的第一个元素
print list[1:3]    # 输出第二个至第三个元素
print list[2:]     # 输出从第三个开始至列表末尾的所有元素
print tinylist * 2 # 输出列表两次
print list + tinylist # 打印组合的列表

```

以上实例输出结果:

```

['runoob', 786, 2.23, 'john', 70.2]

runoob

[786, 2.23]

[2.23, 'john', 70.2]

[123, 'john', 123, 'john']

['runoob', 786, 2.23, 'john', 70.2, 123, 'john']

```

元组: 元组是另一个数据类型, 类似于 List (列表)。元组用 **()** 标识。内部元素用逗号隔开。但是元组不能二次赋值, 相当于只读列表。

```

#!/usr/bin/python

# -*- coding: UTF-8 -*-

tuple = ( 'runoob', 786 , 2.23, 'john', 70.2 )

tinytuple = (123, 'john')

print tuple          # 输出完整元组

```

```
print tuple[0]          # 输出元组的第一个元素

print tuple[1:3]        # 输出第二个至第四个（不包含）的元素

print tuple[2:]         # 输出从第三个开始至列表末尾的所有元素

print tinytuple * 2     # 输出元组两次

print tuple + tinytuple # 打印组合的元组
```

以上实例输出结果:

```
('runoob', 786, 2.23, 'john', 70.2)

runoob

(786, 2.23)

(2.23, 'john', 70.2)

(123, 'john', 123, 'john')

('runoob', 786, 2.23, 'john', 70.2, 123, 'john')
```

注意: 元组是不允许更新的

字典: 字典(dictionary)是除列表以外 python 之中最灵活的内置数据结构类型。列表是有序的对象集合, 字典是无序的对象集合。两者之间的区别在于: 字典当中的元素是通过键来存取的, 而不是通过偏移存取。字典用"{}"标识。字典由索引(key)和它对应的值 value 组成。

```
#!/usr/bin/python

# -*- coding: UTF-8 -*-

dict = {}

dict['one'] = "This is one"

dict[2] = "This is two"

tinydict = {'name': 'runoob', 'code':6734, 'dept': 'sales'}

print dict['one']      # 输出键为'one' 的值

print dict[2]         # 输出键为 2 的值
```

```
print tinydict          # 输出完整的字典
print tinydict.keys()   # 输出所有键
print tinydict.values() # 输出所有值
```

输出为:

```
This is one
This is two
{'dept': 'sales', 'code': 6734, 'name': 'runoob'}
['dept', 'code', 'name']
['sales', 6734, 'runoob']
```

函数:

函数是组织好的，可重复使用的，用来实现单一，或相关联功能的代码段。函数能提高应用的模块性，和代码的重复利用率。Python 提供了许多内建函数，比如 `print()`。也可以自己创建函数，这被叫做用户自定义函数。

自定义函数规则:

- 1、函数代码块以 `def` 关键词开头，后接函数标识符名称和圆括号()。
- 2、任何传入参数和自变量必须放在圆括号中间。圆括号之间可以用于定义参数。
- 3、函数的第一行语句可以选择性地使用文档字符串—用于存放函数说明。
- 4、函数内容以冒号起始，并且缩进。
- 5、`return [表达式]` 结束函数，选择性地返回一个值给调用方。不带表达式的 `return` 相当于返回 `None`。

语法:

```
def functionname( parameters ):
    "函数_文档字符串"
    function_suite
    return [expression]
```

函数定义完成以后，通过另一个函数调用执行。

如：

```
#!/usr/bin/python

# -*- coding: UTF-8 -*-

# 定义函数

def printme( str ):

    "打印任何传入的字符串"

    print str

    return

# 调用函数

printme("我要调用用户自定义函数!")

printme("再次调用同一函数")
```

输出：

```
a=[1,2,3]

a="Runoob"
```

流程控制

比较运算：

== 等于 !=不等于 >大于 >=大于等于 <小于 <=小于等于

if 分支

语法：

```
if 条件表达式:

    满足表达式时进行处理

else:

    不满足表达式时进行处理
```

需要满足多个条件时，可用 **and** 和 **or**。

for 循环

语法:

```
for 变量 in 对象:  
    执行处理
```

模块:

模块是多个对象（方法）的集合，使用 import 进行导入模块，如:

```
import numpy  
  
import numpy as np      #将 numpy 命名为 np  
  
from numpy import sqrt  #从 numpy 模块中导入 sqrt 方法  
  
from numpy.linalg import  #导入 numpy.linalg 中所有方法。
```

常用模块为: numpy (数值计算基础模块)、matplotlib (用于图形绘制)、scipy (数值计算算法扩展模块)、sympy (符号计算模块)、python-control (控制模块)

模型描述

函数名	语法	说明
tf	sys=tf(num,den)	输出传递函数模型
ss	sys=ss(A,B,C,D)	输出状态空间模型
tfdata	[[num]], [[den]]=tfdata(num,den)	取出传递函数的分子多项式、坟墓多项式
ssdata	A,B,C,D=ssdata(sys)	取出状态空间的 A, B, C, D
tf2ss	sysss=tf2ss(systf)	传递函数转换为状态空间
ss2tf	sstf=ss2tf(sysss)	状态空间转换为传递函数
series	sys=series(sys1,sys2)	系统串联
parallel	sys=parallel(sys1,sys2)	系统并联
feedback	sys=feedback(sys1,sys2,sign = -1)	系统负反馈
minreal	sys=minreal(sys)	系统最小实现

模型分析

函数名	语法	说明
pole	p=pole(sys)	输出传递函数的极点
zero	z=zero(sys)	输出产地函数的零点
dcgain	k=dcgain(sys)	输出增益
step	y,t=step(sys,Td)	阶跃响应
impulse	y,t=impulse(sys,Td,x0)	冲激响应
initial	y,t=initial(sys,Td,x0)	零输入响应
lsim	y,t,x0=lsim(sys,Ud,Td,x0)	任意输入时间响应
bode	gain,phase,w=bode(sys,w)	伯德图
nyquist	re,im,w=Nyquist(sys,w)	奈奎斯特图
margin	gm,pm,wpc,wgc=margin(sys)	增益裕量、相位裕量、对应相位裕量的频率， 对应增益裕量的频率
freqresp	g,ph,w=freqresp(sys,wc)	计算特定频率下的幅值和相位

2 Python-control 使用举例

例 1: 在 python control 环境中实现系统串联

```
import control as ctrl

import matplotlib.pyplot as plt

import math

sys1 = ctrl.tf([0,1],[1,1])

sys2 = ctrl.tf([1,1],[1,1,1])

sys3 = ctrl.series(sys1,sys2)      # 两系统串联

print(sys3)
```

输出:

1 2 1

$$\frac{s + 1}{s^3 + 2s^2 + 2s + 1}$$

例 2: 在 python 环境中, 绘制 bode

程序清单:

```
import control as ctrl

import matplotlib.pyplot as plt

import numpy as np

psiList = [0.05,0.2,0.5,0.707,1.0]

sysTfList=[]

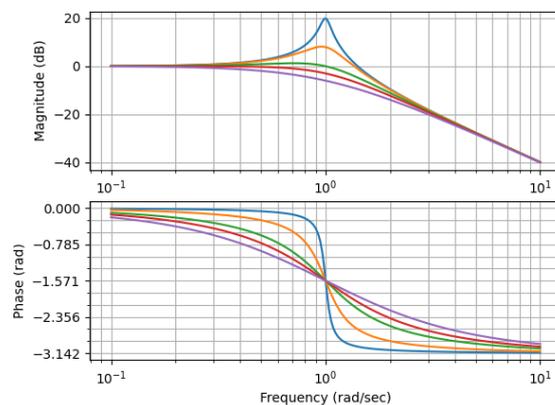
for psi in psiList:

    sysTfList.append(ctrl.tf([1],[1,2*psi,1]))

mag,phase,omega=ctrl.bode(sysTfList,omega=np.logspace(start=1,stop=1,num=200),
dB=True,deg=False)

plt.show()
```

输出:



例 3: python 输出阶跃响应、冲激响应

```
import control as ctrl

import matplotlib.pyplot as plt

sys6 = ctrl.tf(1,[1,0.2,1]) # 二阶系统, 自然谐振频率为 1, 阻尼比为 0.1, 衰减振荡

t1,y1 = ctrl.step_response(sys6)

t2,y2 = ctrl.impulse_response(sys6)

plt.subplot(121)

plt.plot(t1, y1)

plt.title('step response')

plt.ylabel('Amplitude')

plt.xlabel('time (s)')

plt.grid(1)

plt.subplot(122)

plt.plot(t2, y2)

plt.title('impulse response')

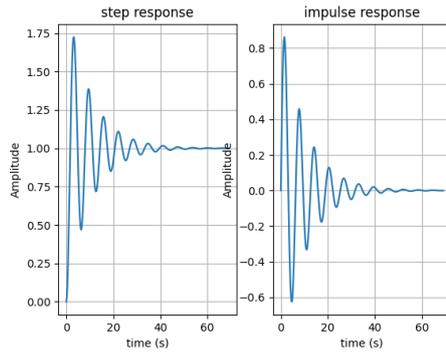
plt.ylabel('Amplitude')

plt.xlabel('time (s)')

plt.grid(1)

plt.show()
```

输出:



例 4: 利用 python, 输出 nyquist 图。

```
import control as ctr

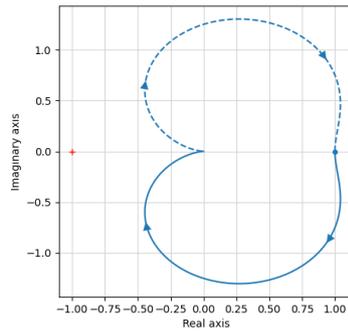
import matplotlib.pyplot as plt

# 使用 tf 生成状态方程
sys = ctr.tf([25],[1,4,25])

# 使用表达式生成状态方程
s = ctr.tf('s')
sys = 25.0/(s**2+4.0*s+25.0)

plt.figure(figsize=(5,5))
ctr.nyquist_plot(sys)
plt.show()
```

输出为:



例 5: 利用 python, 输出根轨迹图

```
import control as ctr

import matplotlib.pyplot as plt

# 使用 tf 生成状态方程
sys = ctr.tf([25],[1,4,25])

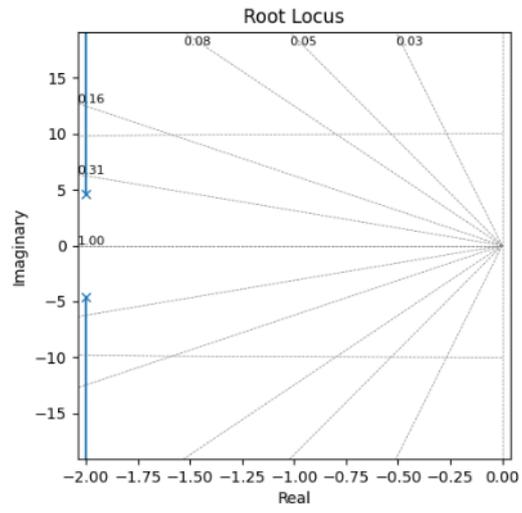
# 使用表达式生成状态方程
s = ctr.tf('s')
sys = 25.0/(s**2+4.0*s+25.0)

plt.figure(figsize=(5,5))

ctr.rlocus(sys)

plt.show()
```

输出为:



第三部分 数字仿真实验

实验六 控制系统的模型转换

实验目的:

掌握控制系统不同模型的表示方法及相互转换。

实验方法:

同一控制系统表示可表示为三种不同的模型：传递函数、零极点增益、状态空间。为分析系统的特性，有必要在各种模型之间进行转换。

传递函数模型

$$H(S) = \frac{\text{num}(s)}{\text{den}(s)} = \frac{b_1 s^m + b_2 s^{m-1} + \cdots + b_{m+1}}{a_1 s^n + a_2 s^{n-1} + \cdots + a_{n+1}}$$

在 MATLAB 中，直接用分子 / 分母的系数表示，即

$$\text{num} = [b_1, b_2, \cdots, b_m] \quad \text{den} = [a_1, a_2, \cdots, a_n]$$

零极点增益模型

$$H(s) = k \frac{(s - z_1)(s - z_2) \cdots (s - z_m)}{(s - p_1)(s - p_2) \cdots (s - p_n)}$$

在 MATLAB 中，用 [z, p, k] 向量组表示，即

$$z = [z_1, z_2, \cdots, z_m];$$

$$p = [p_1, p_2, \cdots, p_n];$$

$$k = [k];$$

状态空间模型 $\dot{x} = ax + bu$

$$y = cx + du$$

在 MATLAB 中，系统可用 (a, b, c, d) 表示

实验内容:

根据系统的传递函数模型

$$H(S) = \frac{num(s)}{den(s)} = \frac{6s + 18}{s^3 + 8s^2 + 17s + 10}$$

求出系统的零极点模型及状态空间模型并判断系统的稳定性。

实验要求：

编制 MATLAB 程序得出零极点模型及状态空间模型并用 MATLAB 画出零极点图。

实验七 控制系统的时域分析

实验目的：

1. 用计算机辅助分析的办法，掌握系统的时域分析方法。
2. 熟悉 Simulink 仿真环境。

二、实验原理及方法：

系统仿真实质上就是对系统模型的求解，对控制系统来说，一般模型可转化成某个微分方程或差分方程表示，因此在仿真过程中，一般以某种数值算法从初态出发，逐步计算系统的响应，最后绘制出系统的响应曲线，进而可分析系统的性能。控制系统最常用的时域分析方法是，当输入信号为单位阶跃和单位冲激函数时，求出系统的输出响应，分别称为单位阶跃响应和单位冲激响应。在 MATLAB 中，提供了求取连续系统的单位阶跃响应函数 `step`，单位冲激响应函数 `impulse`，零输入响应函数 `initial` 等等。

实验内容：

二阶系统，其状态方程模型为

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -0.5572 & -0.7814 \\ 0.7814 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y = [1.9691 \quad 6.4493] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0] u$$

1. 画出系统的单位阶跃响应曲线；
2. 画出系统的冲激响应曲线；
3. 当系统的初始状态为 $x_0=[1,0]$ 时，画出系统的零输入响应；
4. 当系统的初始状态为零时，画出系统斜坡输入响应；

四、实验要求：

1. 编制 MATLAB 程序，画出单位阶跃响应曲线、冲击响应曲线、系统的零输入响应、斜坡输入响应；
2. 在 Simulink 仿真环境中，组成系统的仿真框图，观察单位阶跃响应曲线并记录之。

实验八 控制系统的根轨迹分析

实验目的：

掌握用计算机辅助分析法分析控制系统的根轨迹

熟练掌握 Simulink 仿真环境。

实验原理与方法：

根轨迹是指，当开环系统某一参数（一般来说，这一参数选作开环系统的增益 k ）从零变到无穷大时，闭环系统特征方程的根在 s 平面上的轨迹。因此，从根轨迹，可分析系统的稳定性、稳态性能、动态性能。同时，对于设计系统可通过修改设计参数，使闭环系统具有期望的零极点分布，因此根轨迹对系统设计也具有指导意义。在 MATLAB 中，绘制根轨迹有关的函数有：`rlocus`，`rlocfind`，`pzmap` 等。

实验内容：

一开环系统传递函数为

$$G(s) = \frac{k(s+2)}{(s^2+4s+3)^2}$$

绘制出此闭环系统的根轨迹，并分析系统的稳定性。

四、实验要求：

1. 编制 MATLAB 程序，画出实验所要求根轨迹，求出系统的临界开环增益，并用闭环系统的冲击响应证明之。
2. 在 Simulink 仿真环境中，组成系统的仿真框图，观察临界开环增益时系统单位阶跃响应曲线并记录之。

实验九 控制系统的频域分析

实验目的:

用计算机辅助分析的方法, 掌握频率分析法的三种方法, 即 Bode 图、Nyquist 曲线。

实验原理及方法:

1. Bode(波特)图

设已知系统的传递函数模型:

$$H(s) = \frac{b_1 s^m + b_2 s^{m-1} + \cdots + b_{m+1}}{a_1 s^n + a_2 s^{n-1} + \cdots + a_{n+1}}$$

则系统的频率响应可直接求出:

$$H(j\omega) = \frac{b_1 (j\omega)^m + b_2 (j\omega)^{m-1} + \cdots + b_{m+1}}{a_1 (j\omega)^n + a_2 (j\omega)^{n-1} + \cdots + a_{n+1}}$$

MATLAB 中, 可利用 bode 和 dbode 绘制连续和离散系统的 Bode 图。

2. Nyquist(奈奎斯特)曲线

Nyquist 曲线是根据开环频率特性在复平面上绘制幅相轨迹, 根据开环的 Nyquist 线, 可判断闭环系统的稳定性。

反馈控制系统稳定的充要条件是, Nyquist 曲线按逆时针包围临界点(-1, j0) p 圈, 为开环传递函数位于右半 s 一平面的极点数。在 MATLAB 中, 可利用函数 nyquist 和 dnyquist 绘出连续和离散系统的奈氏曲线。

实验内容:

1. 一系统开环传递函数为

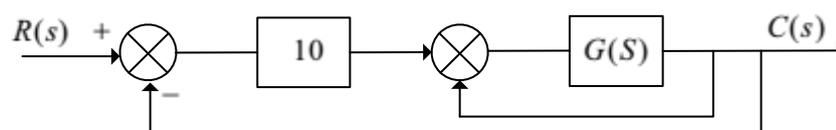
$$H(s) = \frac{50}{(s+1)(s+5)(s-2)}$$

绘制系统的 bode 图, 判断闭环系统的稳定性, 并画出闭环系统的单位冲击响应。

2. 一多环系统

$$G(s) = \frac{16.7s}{(0.8s+1)(0.25s+1)(0.0625s+1)}$$

其结构如图所示



试绘制 Nyquist 频率曲线，并判断稳定性。

实验要求：

1. 编制 MATLAB 程序，画出实验所要求的 Bode 图、Nyquist 图。
2. 在 Simulink 仿真环境中，组成系统的仿真框图，观察单位阶跃响应曲线并记录之。

实验十 控制系统的极点配置

实验目的:

1. 掌握全状态反馈系统的极点配置方法
2. 在 Simulink 仿真环境中, 研究极点配置对系统特性的影响

实验原理与方法:

对一给定控制系统如果其状态完全可控, 则可进行任意极点配置即通过设计反馈增益 K 使闭环系统具有期望的极点。极点配置有二种方法: 第一种方法是采用变换矩阵 T , 使系统具有期望的极点, 从而求出矩阵 K ; 第二种方法基于 Caylay-Hamilton 理论, 通过矩阵多项式 $\phi(\hat{a})$, 可求出 K (这种方法称为 Ackermann 公式)。在 MATLAB 中, 利用控制系统工具箱函数 `place` 和 `acker` 进行极点配置设计。

实验内容:

1. 一被控对象, 其传递函数为

$$G(s) = \frac{10}{(s+1)(s+2)(s+3)}$$

设计反馈控制器 $u=-kx$, 使闭环系统的极点为 $\mu_1 = -2 + j2\sqrt{3}$, $\mu_2 = -2 - j2\sqrt{3}$,

$\mu_3 = -10$ 。

2. 在 Simulink 仿真环境下, 用基本环节组成经过极点配置后的系统, 通过图形观察环节, 观察系统的各点响应。

四、实验要求:

1. 编制 MATLAB 程序, 按希望的闭环极点, 求出 K 向量。
2. 在 Simulink 仿真环境中, 组成系统的仿真框图, 记录实验中的输出响应曲线。

附录

DS1102E 数字示波器简介

DS1102E 示波器向用户提供简单而功能明晰的前面板，以进行所有的基本操作。各通道的标度和位置旋钮提供了直观的操作，完全符合传统仪器的使用习惯，用户不必花大量的时间去学习和熟悉示波器的操作，即可熟练使用。为加速调整，便于测量，用户可直接按 AUTO 键，立即获得适合的波形显现和档位设置。

除易于使用之外，DS1102E 示波器还具有更快完成测量所需要的高性能指标和强大功能。通过 1GSa/s 的实时采样和 25GSa/s 的等效采样，可在 DS1102E 上观察更快的信号。强大的触发和分析能力使其易于捕获和分析波形。清晰的液晶显示和数学运算功能，便于用户更快更清晰地观察和分析信号问题。

DS1102E 示波器有以下特点：

双模拟通道，每通道带宽 100M；高清晰彩色液晶显示系统，320x234 分辨率；支持即插即用闪存式 USB 存储设备以及 USB 接口打印机，并可通过 USB 存储设备进行软件升级；模拟通道的波形亮度可调；自动波形、状态设置 (AUTO)；波形、设置、CSV 和位图文件存储以及波形和设置再现；精细的延迟扫描功能，轻易兼顾波形细节与概貌；自动测量 20 种波形参数；自动光标跟踪测量功能；独特的波形录制和回放功能；内嵌 FFT；实用的数字滤波器，包含 LPF, HPF, BPF, BRF；Pass/Fail 检测功能，光电隔离的 Pass/Fail 输出端口；多重波形数学运算功能；边沿、视频、斜率、脉宽、交替、触发功能独一无二的可变触发灵敏度，适应不同场合下特殊测量要求。

DS1102E 面板上包括旋钮和功能按键。旋钮的功能与其它示波器类似。显示屏右侧的一列 5 个灰色按键为菜单操作键（自上而下定义为 1 号至 5 号），可以设置当前菜单的不同选项；其它按键为功能键，可以进入不同的功能菜单或直接获得特定的功能应用。

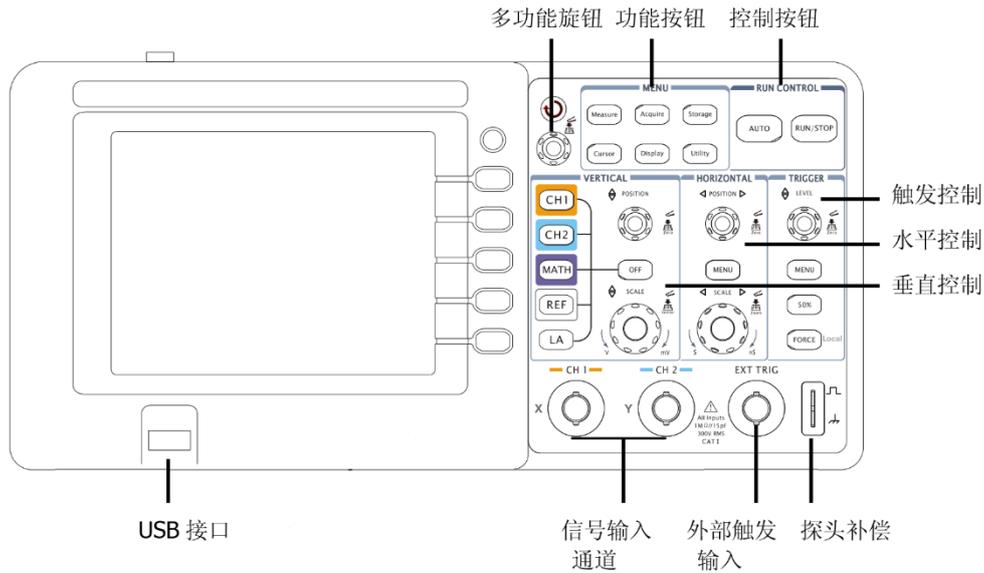


图 1 面板操作说明图

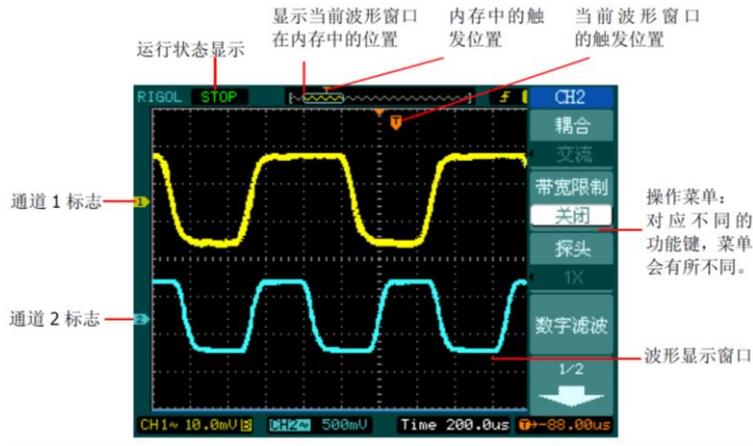


图 2 显示界面说明图

下面分别从功能检查、探头补偿、垂直系统、水平系统、触发系统、通道耦合、通道带宽限制、数字滤波、减少信号上的噪声、X-Y 功能应用进行介绍。

1、功能检查

1.1、接通电源：接通电源后，仪器执行所有自检项目，并确认通过自检按 STORAGE 按钮，用菜单操作键从顶部菜单框中选择存贮类型，然后调出出厂设置菜单框

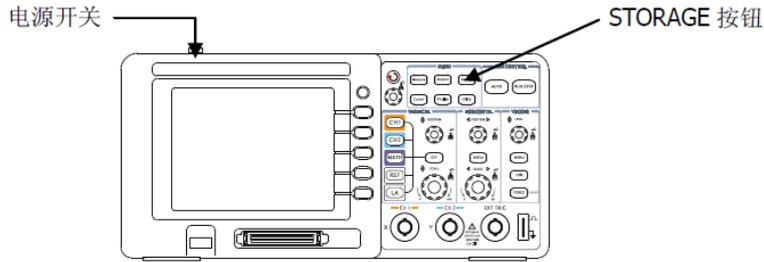


图 3 上电后检查

1.2、用示波器探头将信号接入通道 1 (CH1)：将探头上的开关设定为 10X (图 1-6)，并将示波器探头与通道 1 连接。将探头连接器上的插槽对准 CH1 同轴电缆插接口 (BNC) 上的插口并插入，然后向右旋转以拧紧探头。

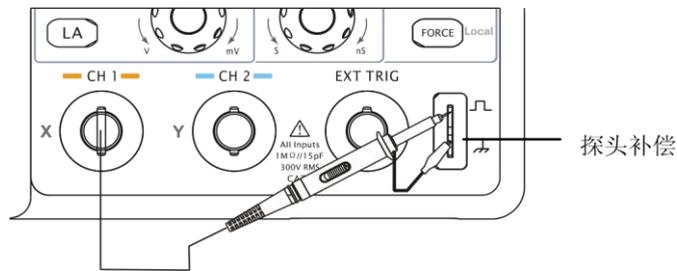


图 4 探头补偿连接

1.3、示波器需要输入探头衰减系数。此衰减系数改变仪器的垂直档位比例，从而使得测量结果正确反映被测信号的电平。(默认的探头菜单衰减系数设定值为 1X。) 设置探头衰减系数的方法如下：按 CH 功能键显示通道 1 的操作菜单，应用与探头项目平行的 3 号菜单操作键，选择与您使用的探头同比例的衰减系数。此时设定应为 10X。



图 5 设定探头上的系数



图 6 设定菜单中的系数

1.4、把探头端部和接地夹接到探头补偿器的连接器的上。按 AUTO 自动设置) 按钮。几秒钟内，可见到方波显示，方波的幅值为 1V，频率为 1KHZ。

1.5、以同样的方法检查通道 2 (CH2)。按 OFF 功能按钮或再次按下功能按钮以关闭通道 1，按 CH2 功能按钮以打开通道 2,重复步骤 3 和步骤 4。

注意：探头补偿连接器输出的信号仅作探头补偿调整之用，不可用于校准。

2、探头补偿

在首次将探头与任一输入通道连接时，进行此项调节，使探头与输入通道相配。未经补偿或补偿偏差的探头会导致测量误差或错误。若调整探头补偿，请按如下步骤：

2.1、将探头菜单衰减系数设定为 10X,将探头上的开关设定为 10X,并将示波器探头与通道 1 连接。如使用探头钩形头，应确保与探头接触紧密。将探头端部与探头补偿器的信号输出连接器相连，基准导线夹与探头补偿器的地线连接器相连，打开通道 1,然后按 AUTO。

2.2、检查所显示波形的形状。



图 7 探头补偿调节

2.3、如必要，用非金属质地的改锥调整探头上的可变电容，直到屏幕显示的波形如上图“补偿正确”。

2.4、必要时，重复步骤。

3、垂直系统

如下图所示，在垂直控制区（VERTICAL）有一系列的按键、旋钮。

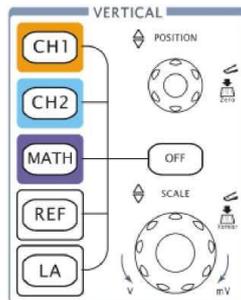


图 8 垂直控制系统

3.1、使用垂直旋钮在波形窗口居中显示信号。

垂直 POSITION 旋钮控制信号的垂直显示位置。当转动垂直 POSITION 旋钮时，指示通道地（GROUND）的标识跟随波形而上下移动。

测量技巧

如果通道耦合方式为 DC，您可以通过观察波形与信号地之间的差距来快速测量信号的直流分量。

如果耦合方式为 AC，信号里面的直流分量被滤除。这种方式方便您用更高的灵敏度显示信号的交流分量。

双模拟通道垂直位置恢复到零点快捷键旋动垂直 POSITION 旋钮不但可以改变通道的垂直显示位置，更可以通过按下该旋钮作为设置通道垂直显示位置恢复到零点的快捷键。

3.2、改变垂直设置，并观察因此导致的状态信息变化。

可以通过波形窗口下方的状态栏显示的信息，确定任何垂直档位的变化。

转动垂直 SCALE 旋钮改变“Volt/div(伏/格)”垂直档位，可以发现状态栏对应通道的档位显示发生了相应的变化。

按 CH1、CH2、MATCH、REF 屏幕显示对应通道的操作菜单、标志、波形和档位状态信息。按 OFF 按键关闭当前选择的通道。

Coarse/Fine(粗调/微调) 快捷键

可通过按下垂直旋钮作为设置输入通道的粗调/微调状态的快捷键，然后调节该旋钮即可粗调/微调垂直档位。

4、设置垂直系统

每个通道有独立的垂直菜单。每个项目都按不同的通道单独设置。

按 CH1 或 CH2 功能按键，系统显示 CH1 或 CH2 通道的操作菜单，说明见下表：

CH1	功能菜单	设定	说明	CH1			
耦合	耦合	交流	阻挡输入信号的直流成分。 通过输入信号的交流和直流成分。断开输入信号。	2/2	(上一页)	2/2	返回上一页菜单（以下均同，不再说明）
直流		接地					
带宽限制	带宽限制	打开	限制带宽至20MHz。以减少显示噪音。满带宽。	档位调节	粗调	微调	粗调按1-2-5进制设定垂直灵敏度。 微调则在粗调设置范围之间进一步细分，以改善垂直分辨率。
关闭		关闭					
探头	探头	1X	根据探头衰减因数选取其中一个值，以保持垂直标尺读数准确。	反相	反相	打开	打开波形反向功能。 波形正常显示。
1X		5X					
数字滤波		10X					
		50X					
		100X					
		500X					
1000X							
1/2	数字滤波	设置数字滤波（见表2—4）	关闭	关闭	关闭		
(下一页)	1/2	进入下一页菜单（以下均同，不再说明）					

图9 通道操作菜单

5、水平系统

如下图所示，在水平控制区（HORIZONTAL）有一个按键、两个旋钮。

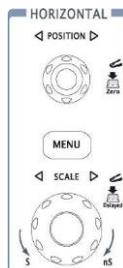


图 10 水平控制区

5.1、使用水平旋钮改变水平档位设置，并观察因此导致的状态信息变化。转动水平 SCALE 旋钮改变“s/div(秒/格)”水平档位，可以发现状态栏对应通道的档位显示发生了相应的变化。水平扫描速度从 2ns 至 50s，以 1-2-5 的形式步进。

Delayed (延迟扫描) 快捷键

水平 SCALE 旋钮不但可以通过转动调整“s/div (秒/格)”，更可以按下切换到延迟扫描状态。

5.2、使用水平旋钮调整信号在波形窗口的水平位置。

水平旋钮控制信号的触发位移。当应用于触发位移时，转动水平 POSITION 旋钮时，可以观察到波形随旋钮而水平移动。

触发点位移恢复到水平零点快捷键

水平 POSITION 旋钮不但可以通过转动调整信号在波形窗口的水平位置，更可以按下该键使触发位移（或延迟扫描位移）恢复到水平零点处。

5.3、按 MENU 按钮，显示 TIME 菜单。在此菜单下，可以开启/关闭延迟扫描或切换 Y—T、X—Y 和 ROLL 模式，还可以设置水平触发位移复位。

名词解释

触发位移：指实际触发点相对于存储器中点的位置。转动水平 POSITION 旋钮，可水平移动触发点

6、触发系统

如下图所示，在触发控制区 (TRIGGER)有一个旋钮、三个按键。

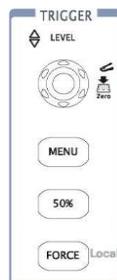


图 11 触发控制区

6.1、使用 LEVEL 旋钮改变触发电平设置。

转动 LEVEL 旋钮，可以发现屏幕上出现一条桔红色的触发线以及触发标志，随旋钮转动而上下移动。停止转动旋钮，此触发线和触发标志会在约 5 秒后消失。在移动触发线的同时，可以观察到在屏幕上触发电平的数值发生了变化。

触发电平恢复到零点快捷键

旋转垂直 LEVEL 旋钮不但可以改变触发电平值，更可以通过按下该旋钮作为设置触发电平恢复到零点的快捷键。

6.2、使用 MENU 调出触发操作菜单（见下图），改变触发的设置，观察由此造成的状态变化。

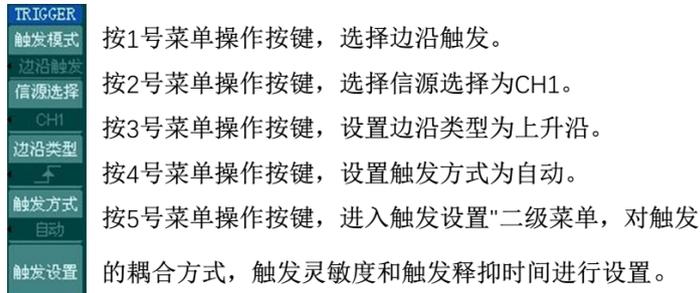


图 12 触发菜单

注：改变前三项的设置会导致屏幕右上角状态栏的变化。

6.3、按 50%按钮，设定触发电平在触发信号幅值的垂直中点。

6.4、按 FORCE 按钮:强制产生一触发信号，主要应用于触发方式中的普通和单次模式。

名词解释

触发释抑：指重新启动触发电路的时间间隔。旋转多功能旋钮，可设置触发释抑时间。

7、通道耦合

7.1、以 CH1 通道为例，被测信号是一含有直流偏置的正弦信号。

按 CH1->耦合->交流，设置为交流耦合方式。被测信号含有的直流分量被阻隔。波形显示如下图所示。

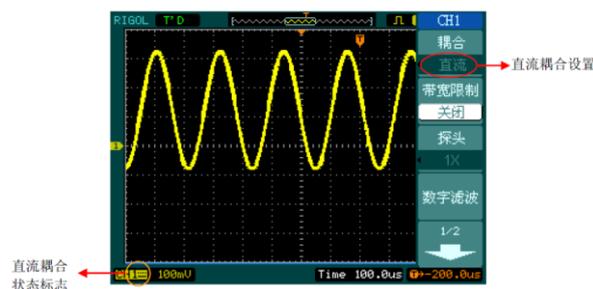


图 13 交流耦合设置

按 CH1->耦合->直流，设置为直流耦合方式。被测信号含有的直流分量和交流分量都可以通过。波形显示如下图所示。

按 CH1->耦合->接地，设置为接地方式。被测信号含有的直流分量和交流分量都被阻隔。波形显示如下图所示。

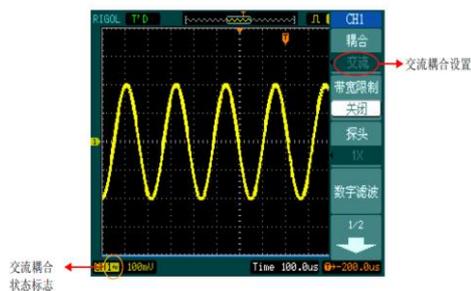


图 14 直流耦合设置图

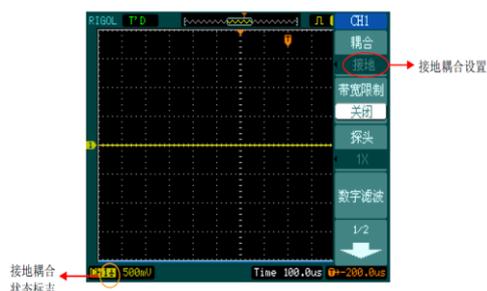


图 15 接地耦合设置

8、通道带宽限制

以 CH1 通道为例，被测信号是一含有高频振荡的脉冲信号。

按 CH1->带宽限制->关闭，设置带宽限制为关闭状态。被测信号含有的高频分量可以通过。波形显示如下图所示。

按 CH1->带宽限制->打开，设置带宽限制为打开状态。被测信号含有的大于 20MHz 的高频分量被阻隔。波形显示如下图所示。

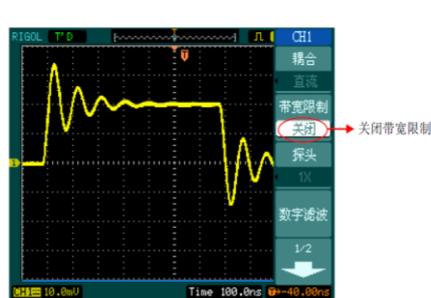


图 15 关闭带宽限制

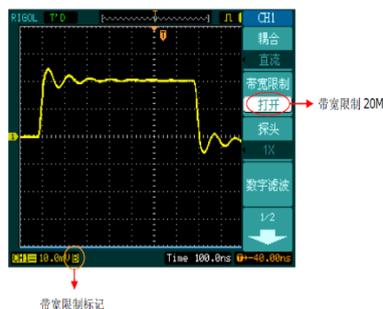


图 16 打开带宽限制

9、数字滤波

按 CH1->数字滤波，系统显示 FILTER 数字滤波功能菜单，旋动多功能旋钮设置频率上限和下限，设定滤波器的带宽范围。说明见表 2:

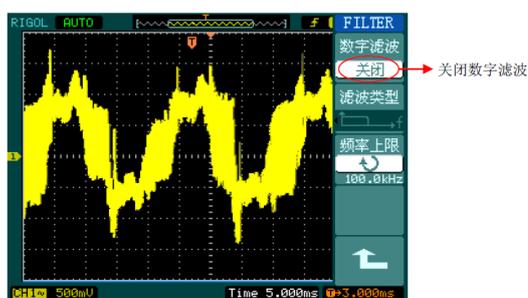


图 17 关闭数字滤波



图 18 打开数字滤波



功能菜单	设定	说明
数字滤波	关闭 打开	关闭数字滤波器 打开数字滤波器
滤波类型		设置滤波器为低通滤波 设置滤波器为高通滤波 设置滤波器为带通滤波 设置滤波器为带阻滤波
频率上限	 < 上限频率 >	多功能旋钮 () 设置频率上限
频率下限	 < 下限频率 >	多功能旋钮 () 设置频率下限
		返回上一级菜单 (以下均同, 不再说明)

表 2 数字滤波设置菜单

10、减少信号上的随机噪声

如果被测试的信号上叠加了随机噪声，您可以通过调整本示波器的设置，滤除或减小噪声，避免其在测量中对本体信号的干扰（波形见下图）。

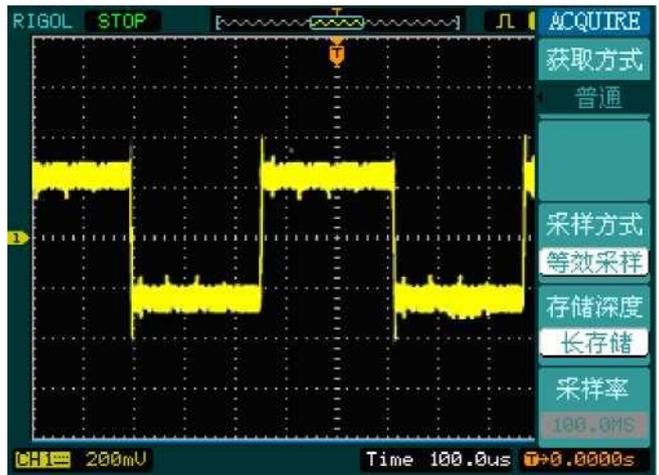


图 19 叠加噪声的波形

操作步骤如下：

- 10.1、如前例设置探头和 CH1 通道的衰减系数。
- 10.2、连接信号使波形在示波器上稳定地显示。
- 10.3、通过设置触发耦合改善触发。

- 1)、按下触发 TRIGGER 控制区域 MENU 按钮，显示触发设置菜单。
- 2)、触发设置->耦合->选择低频抑制或高频抑制。

低频抑制是设定一高通滤波器，可滤除 8kHz 以下的低频信号分量，允许高频信号分量通过。

高频抑制是设定一低通滤波器，可滤除 150kHz 以上的高频信号分量（如 FM 广播信号），允许低频信

号分量通过。通过设置低频抑制或高频抑制可以分别抑制低频或高频噪声，以得到稳定的触发。

10.4、通过设置采样方式和调整波形亮度减少显示噪声。

(1)如果被测信号上叠加了随机噪声，导致波形过粗。可以应用平均采样方式，去除随机噪声的显示，使波形变细，便于观察和测量。取平均值后随机噪声被减小而信号的细节更易观察。

具体的操作是：按面板 MENU 区域的 ACQUIRE 按钮，显示采样设置菜单。按 1 号菜单操作键设置获取方式为平均状态，然后按 2 号菜单操作键调整平均次数，依次由 2 至 256 以 2 倍数步进，直至波形的显示满足观察和测试要求。

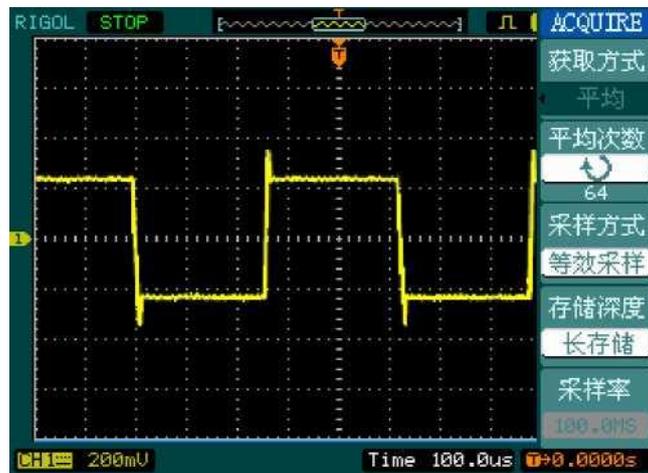


图 20 减少噪声后的波形

(2)减少显示噪声也可以通过减低波形亮度来实现。

注意：使用平均采样方式会使波形显示更新速度变慢，这是正常现象。

11、X-Y 功能的应用

查看两通道信号的相位差：测试信号经过一电路网络产生的相位变化。

将示波器与电路连接，监测电路的输入输出信号。

欲以 X—Y 坐标图的形式查看电路的输入输出，请按如下步骤操作：

- 1、将探头菜单衰减系数设定为 10X，并将探头上的开关设定为 10X。
- 2、将通道 1 的探头连接至网络的输入，将通道 2 的探头连接至网络的输出。
- 3、若通道未被显示，则按下 CH1 和 CH2 菜单按钮。
- 4、按下 AUTO(自动设置)按钮。
- 5、调整垂直旋钮使两路信号显示的幅值大约相等。

6、按下水平控制区域的 MENU 菜单按钮以调出水平控制菜单。

7、按下时基菜单框按钮以选择 X—Y。

8、示波器将以李沙育 (Lissajous)图形模式显示网络的输入输出特征。9、调整垂直 SCALE、垂直 POSITION 和水平 SCALE 旋钮使波形达到最佳效果。

应用椭圆示波图形法观测并计算出相位差。(见下图)

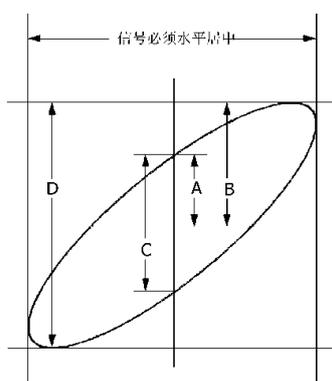


图 21 椭圆示波图形法

根据 $\sin\theta = A/B$ 或 C/D , 其中 θ 为通道间的相差角, A、B、C、D 的定义见上图。因此可以得出相差角, 即: $\theta = \pm \arcsin(A/B)$ 或 $\pm \arcsin(C/D)$ 。

如果椭圆的主轴在 I、III 象限内, 那么所求得相位差角应在 I、IV 象限内, 即在 $(0 \sim \pi/2)$ 或 $(3\pi/2 \sim 2\pi)$ 内。如果椭圆的主轴在 II、IV 象限内, 那么所求得相位差角应在 II、III 象限内, 即在 $(\pi/2 \sim \pi)$ 或 $(\pi \sim 3\pi/2)$ 内。

名词解释

信源：触发可从多种信源得到：输入通道（CH1、CH2），外部触发（EXT），ACLine（市电）。

输入通道：最常用的触发信源是输入通道（可任选一个）。被选中作为触发信源的通道，无论其输入是否被显示，都能正常工作。

外部触发：这种触发信源可用于在两个通道上采集数据的同时在第三个通道上触发。例如，可利用外部时钟或来自待测电路的信号作为触发信源。EXT 触发源都使用连接至 EXTRIG 接头的外部触发信号。

EXT 可直接使用信号，您可在信号触发电平范围在-1.2V 至+1.2V 时使用 EXT。

AC Line：即交流电源。这种触发信源用来显示信号与动力电，如照明设备和动力提供设备之间的关系。示波器将产生触发，无需人工输入触发信号。

在使用交流电源作为触发信源时，触发电平设定为 0 伏，不可调节。

触发方式：决定示波器在无触发事件情况下的行为方式。本示波器提供三种触发方式：自动，普通和单次触发。

自动触发：这种触发方式使得示波器即使在没有检测到触发条件的情况下也能采样波形。当示波器在一定等待时间（该时间可由时基设置决定）内没有触发条件发生时，示波器将进行强制触发。

当强制进行无效触发时，示波器虽然显示波形，但不能使波形同步，则显示的波形将不稳定。当有效触发发生时，显示器上的波形是稳定的。

可用自动方式来监测幅值电平等可能导致波形显示不稳定的因素，如动力供应输出等。注意：在扫描波形设定在 50ms/div 或更慢的时基上时，自动方式允许没有触发信号。

普通触发：示波器在普通触发方式下只有当触发条件满足时才能采样到波形。在没有触发时，示波器将显示原有波形而等待触发。

单次触发：在单次触发方式下，用户按一次“运行”按钮，示波器等待触发，当示波器检测到一次触发时，采样并显示一个波形，采样停止。

耦合：

触发耦合决定信号的何种分量被传送到触发电路。耦合类型包括直流，交流，低频抑制和高频抑制。

直流：让信号的所有成分通过。

交流：阻挡直流成分并衰减 10Hz 以下信号。

低频抑制：阻挡直流成分并衰减低于 8kHz 的低频成分。 高频抑制：衰减超过 150kHz 的高频成分。

预触发/延迟触发：触发事件以前/后采样的数据。

触发位置通常设定在屏幕的水平中心。在全屏显示情况下，您可以观察到 6 格的预触发和延迟信息。您可以旋转水平调节波形的水平位移，查看更多的预触发信息，或者最多触发 1s 后的延迟触发信息。

通过观察预触发数据，可以了解触发以前的信号情况。例如捕捉到电路产生的毛刺，通过观察和分析预触发数据，可能会查出毛刺产生的原因。

可变触发灵敏度：

为了排除现实世界信号噪声的影响，得到稳定的触发，触发电路引入了迟滞。范围是 0.1div—1.0div。

即当设置为 1.0div 时，触发电路对于任何峰-峰幅度 1.0div 的信号，不做响应，从而排除噪声的影响。

电压参数的自动测量

DS1102E 可以自动测量的电压参数包括峰峰值、最大值、最小值、平均值、均方根值、顶端值、低端值。下图表述了一系列电压参数的物理意义。

峰峰值 (Vpp): 波形最高点波峰至最低点的电压值。

最大值 (Vmax): 波形最高点至 GND(地) 的电压值。

最小值 (Vmin): 波形最低点至 GND(地) 的电压值。

幅值 (Vamp): 波形顶端至底端的电压值。

顶端值 (Vtop): 波形平顶至 GND(地) 的电压值。

底端值 (Vbase): 波形平底至 GND(地) 的电压值。

过冲 (Overshoot): 波形最大值与顶端值之差与幅值的比值。

预冲 (Preshoot): 波形最小值与底端值之差与幅值的比值。

平均值 (Average): 单位时间内信号的平均幅值。

均方根值 (Vrms): 即有效值。依据交流信号在单位时间内所换算产生的能量, 对于产生等值能量的直流电压, 即均方根值。

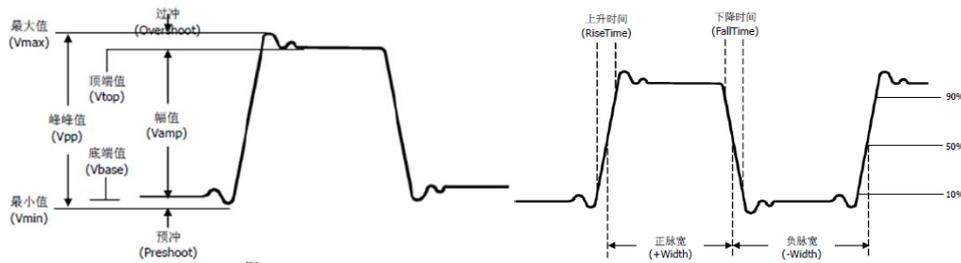


图 22 电压参数的自动测量图

图 23 时间参数的自动测量

时间参数的自动测量

DS1102E 可以自动测量信号的频率、周期、上升时间、下降时间、正脉宽、负脉宽、延迟 1 今 2f、延迟 1 今 2t、正占空比、负占空比十种时间参数的自动测量。上升时间 (RiseTime): 波形幅度从 10% 上升至 90% 所经历的时间。下降时间 (FallTime): 波形幅度从 90% 下降至 10% 所经历的时间。

正脉宽 (+Width): 正脉冲在 50% 幅度时的脉冲宽度。

负脉宽 (-Width): 负脉冲在 50% 幅度时的脉冲宽度。

延迟 1+2f (Delay1 今 2f): 通道 1、2 相对于上升沿的延时。

延迟 1+2 士 (Delay1+2 士): 通道 1、2 相对于下降沿的延时。

正占空比 (+Duty): 正脉宽与周期的比值。

负占空比 (-Duty): 负脉宽与周期的比值。

DG1022U 双通道函数/任意波形发生器简介

DG1022U 双通道函数/任意波形发生器使用直接数字合成 (DDS) 技术, 可生成稳定、精确、纯净和低失真的正弦信号。它还能提供 5MHz、具有快速上升沿和下降沿的方波。另外还具有高精度、宽频带的频率测量功能。DG1022U 实现了易用性、优异的技术指标及众多功能特性的完美结合, 可帮助用户更快地完成工作任务。

DG1022U 系列双通道函数/任意波形发生器向用户提供简单而功能明晰的前面板。人性化的键盘布局和指示以及丰富的接口, 直观的图形用户操作界面, 内置的提示和上下文帮助系统极大地简化了复杂的操作过程, 用户不必花大量的时间去学习和熟悉信号发生器的操作, 即可熟练使用。内部 AM、FM、PM、FSK 调制功能使仪器能够方便的调制波形, 而无需单独的调制源。

性能特点:

DDS 直接数字合成技术, 得到精确、稳定、低失真的输出信号; 双通道输出, 可实现通道耦合, 通道复制; 输出 5 种基本波形, 内置 48 种任意波形; 可编辑输出 14-bit、4k 点的用户自定义任意波形; 100MSa/s 采样率;

频率特性:

正弦波: 1 μ Hz 到 20MHz; 方波: 1 μ Hz 到 5MHz; 锯齿波: 1 μ Hz 到 150kHz; 脉冲波: 500 μ Hz 到 3MHz; 白噪声: 5MHz 带宽 (-3dB); 任意波形: 1 μ Hz 到 5MHz

幅度范围: 2mVpp \sim 10Vpp(50 Ω)、4mVpp \sim 20Vpp(高阻)

具有丰富的调制功能, 输出各种调制波形: 调幅 (AM)、调频 (FM)、调相 (PM)、二进制频移键控 (FSK)、线性和对数扫描 (Sweep) 及脉冲串 (Burst) 模式;

丰富的输入输出: 外接调制源, 外接基准 10MHz 时钟源, 外触发输入, 波形输出, 数字同步信号输出;

高精度、宽频带频率计: 测量功能: 频率、周期、占空比、正/负脉冲宽度频率范围:
100mHz \sim 200MHz (单通道)

支持即插即用 USB 存储设备, 并可通过 USB 存储设备存储、读取波形配置参数及用户自定义任意波形, 以及进行软件升级;

标准配置接口: USB Host&Device;

图形化界面可以对信号设置进行可视化验证；

中英文嵌入式帮助系统；

支持中英文输入。

DG1022U 面板介绍

DG1022U 向用户提供简单而功能明晰的前面板，如图 1-3 所示，前面板上包括各种功能按键、旋钮及菜单软键，您可以进入不同的功能菜单或直接获得特定的功能应用。

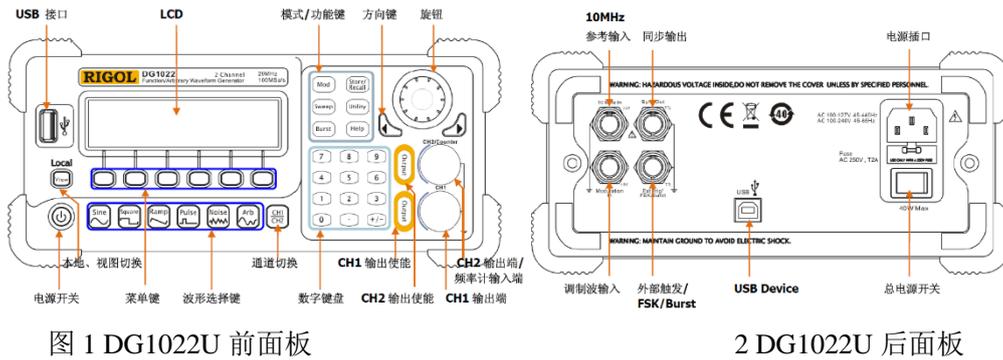


图 1 DG1022U 前面板

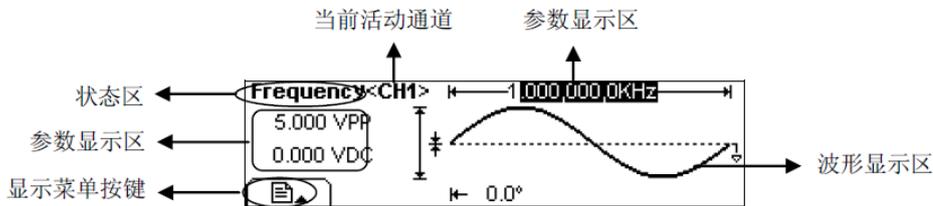
2 DG1022U 后面板

DG1022U 的用户界面

DG1022U 双通道函数/任意波形发生器提供了 3 种界面显示模式：单通道常规模式、单通道图形模式及双通道常规模式。这 3 种显示模式可通过前面板左侧的 View 按键切换。用户可通过 ^ 来切换活动通道，以便于设定每通道的参数及观察、比较波形。



图 1-5 单通道常规显示模式图



1-6 单通道图形显示模式



图 1-7 双通道常规显示模式

按键表示说明：

本书对按键的标识用加边框的字符表示，如 Sine 代表前面板上一个标注着“Sine”字符的功能键，菜单软键的标识用带阴影的字符表示，如频率表示 sine 菜单中的“频率”选项。

波形设置

如下图所示，在操作面板左侧下方有一系列带有波形显示的按键，它们分别是：正弦波、方波、锯齿波、脉冲波、噪声波、任意波，此外还有两个常用按键：通道选择和视图切换键。

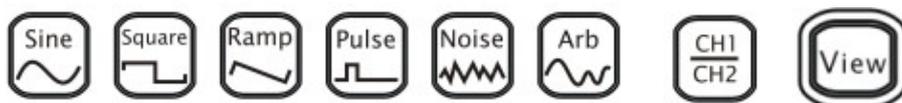


图 1-8 按键选择

1、使用 Sine 按键，波形图标变为正弦信号，并在状态区左侧出现“Sine”字样。

DG1022U 可输出频率从 1 Hz 到 20MHz 的正弦波形。通过设置频率/周期、幅值/高电平、偏移/低电平、相位，可以得到不同参数值的正弦波。



图 1-9 正弦波常规显示界面

上图所示正弦波使用系统默认参数：频率为 1kHz，幅值为 5.0VPP，偏移量为 0VDC，初始相位为 0°。

2、使用 Square 按键，波形图标变为方波信号，并在状态区左侧出现“Square”字样。

DG1022U 可输出频率从 1 μ Hz 到 5MHz 并具有可变占空比的方波。通过设置频率/周期、幅值/高电平、偏移/低电平、占空比、相位，可以得到不同参数值的方波。



图 1-10 方波常规显示界面

图 1-10 所示方波使用系统默认参数:频率为 1kHz, 幅值为 5.0VPP, 偏移量为 0Vdc, 占空比为 50%, 初始相位为 0°。

3、使用 Ramp 按键, 波形图标变为锯齿波信号, 并在状态区左侧出现“Ramp”字样。DG1022U 可输出频率大小从 1 μHz 到 150kHz 并具有可变对称性的锯齿波波形。通过设置频率/周期、幅值/高电平、偏移/低电平、对称性、相位, 可以得到不同参数值的锯齿波。



图 1-11 锯齿波常规显示界面

图 1-11 所示锯齿波使用系统默认参数: 频率为 1kHz, 幅值为 5.0Vpp, 偏移量为 0Vdc, 对称性为 50%, 初始相位为 0°。

4、使用 Pulse 按键, 波形图标变为脉冲波信号, 并在状态区左侧出现“Pulse”字样。DG1022U 可输出频率从 500 μHz 到 3MHz 并具有可变脉冲宽度的脉冲波形。通过设置频率/周期、幅值/高电平、偏移/低电平、脉宽/占空比、延时, 可以得到不同参数值的脉冲波。



图 1-12 脉冲波常规显示界面

图 1-12 所示脉冲波形使用系统默认参数: 频率为 1kHz, 幅值为 5.0VPP, 偏移量为 0VDC, 脉宽为 500 叫占空比为 50%, 延时为 0s。

5、使用 Noise 按键, 波形图标变为噪声信号, 并在状态区左侧出现“Noise”字样。DG1022U 可输出带宽为 5MHz 的噪声。通过设置幅值/高电平、偏移/低电平, 可以得到不同参数值的噪声信号。



图 1-13 噪声波形常规显示界面

图 1-13 所示波形为系统默认的信号参数：幅值为 5.0Vpp，偏移量为 0Vdc

6、使用 Arb 按键，波形图标变为任意波信号，并在状态区左侧出现“Arb”字样。DG10×2 可输出最多 4K 个点和最高 5MHz 重复频率的任意波形。通过设置频率/周期、幅值/高电平、偏移/低电平、相位，可以得到不同参数值的任意波信号。

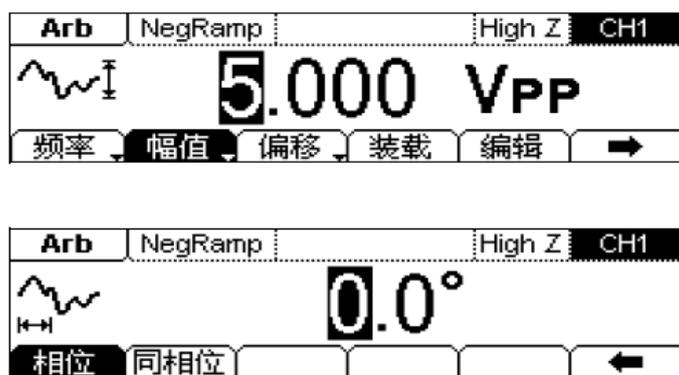


图 1-14 任意波形常规显示界面

图 1-14 所示 NegRamp 倒三角波形使用系统默认参数：频率为 1kHz，幅值为 5.0Vpp，偏移量为 0Vdc，相位为 0°。

7、使用 CH1/CH2 键切换通道，当前选中的通道可以进行参数设置。在常规和图形模式下均可以进行通道切换，以使用户观察和比较两通道中的波形。

8、使用 View 键切换视图，使波形显示在单通道常规模式。单通道图形模式、双通道常规模式之间切换。此外，当仪器处于远程模式，按下该键可以切换到本地模式。

输出设置

如下图 1-15 所示，在前面板右侧有两个按键，用于通道输出、频率计输入的控制。下面的说明将引导您逐步熟悉这些功能。

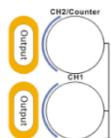


图 1-15 通道输出、频率计输入

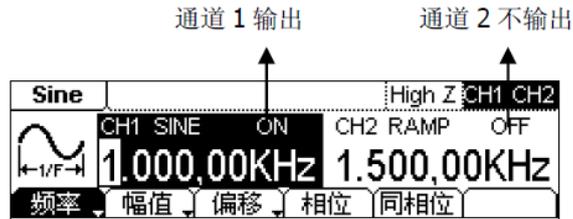


图 1-16 通道输出控制

1、使用 Output 按键，启用或禁用前面板的输出连接器输出信号。如图 1-16 所示，已按下 Output 键的通道显示“ON”且 Output 点亮。

2、在频率计模式下，CH2 对应的 Output 连接器作为频率计的信号输入端，CH2 自动关闭，禁用输出。

数字输入的使用

如下图 1-21 所示，在前面板上有两组按键，分别是左右方向键和旋钮、数字键盘。下面的说明将逐渐引导您熟悉数字输入功能的使用。

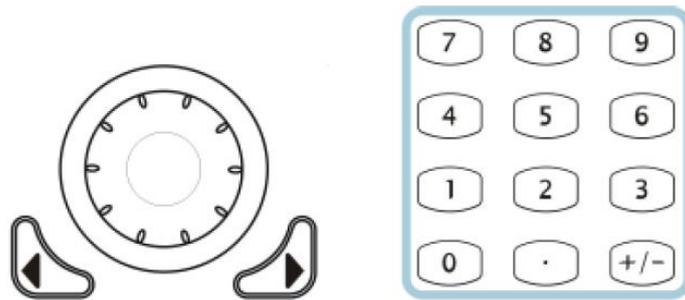


图 1-21 前面板的数字输入

1、使用左右方向键，用于数值不同数位的切换；使用旋钮，用于改变波形参数的某数位数值的大小，旋钮的输入范围是 0~9,旋钮顺时针旋一格，数值增 1。

2、使用数字键盘，用于波形参数值的设置，直接改变参数值的大小。设置正弦波

使用 Sine 按键，常规显示模式下，在屏幕下方显示正弦波的操作菜单，左上角显示当前波形名称。通过使用正弦波的操作菜单，对正弦波的输出波形参数进行设置。

设置正弦波参数主要包括：频率/周期，幅值/高电平，偏移/低电平，相位。通过改变这些参数，得到不同的正弦波。如图 2-1 示，在操作菜单中，选中频率，光标位于参数显示区的频率参数位置，用户可在此位置通过数字键盘、方向键或旋钮对正弦波的频率值进行修改。操作菜单：通过软键控制使用



图 2-1 正弦波参数值设置显示界面

表 2-1 Sine 波形的菜单说明

功能菜单	设定	说明
频率/ 周期	/	设置波形频率或周期
幅值/ 高电平	/	设置波形幅值或高电平
偏移/ 低电平	/	设置波形偏移量或低电平
相位	/	设置正弦波的起始相位

提示说明

操作菜单中的同相位专用于使能双通道输出时相位同步，单通道波形无需配置此项。

设置输出频率/周期

1、按 Sine→频率/周期→频率，设置频率参数值。

屏幕中显示的频率为上电时的默认值，或者是预先选定的频率。在更改参数时，如果当前频率值对于新波形是有效的，则继续使用当前值。若要设置波形周期，则再次按频率/周期软键，以切换到周期软键（当前选项为反色显示）。

2、输入所需的频率值。使用数字键盘，直接输入所选参数值，然后选择频率所需单位，按下对应于所需单位的软键。也可以使用左右键选择需要修改的参数值的数位，使用旋钮改变该数位值的大小。

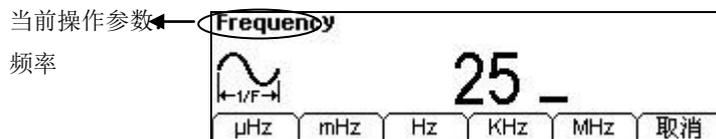


图 2-2 设置频率的参数值

提示说明

当使用数字键盘输入数值时，使用方向键的左键退位，删除前一位的输入，修改输入的数值。

当使用旋钮输入数值时，使用方向键选择需要修改的位数，使其反色显示，然后转动旋钮，修改此位数字，获得所需要的数值。

设置输出幅值

- 1、按 Sine→幅值/高电平→幅值，设置幅值参数值。屏幕显示的幅值为上电时的默认值，或者是预先选定的幅值。在更改参数时，如果当前幅值对于新波形是有效的，则继续使用当前值。若要使用高电平和低电平设置幅值，再次按幅值/高电平或者偏移/低电平软键，以切换到高电平和低电平软键（当前选项为反色显示）。
- 2、输入所需的幅值。使用数字键盘或旋钮，输入所选参数值，然后选择幅值所需单位，按下对应于所需单位的软键。

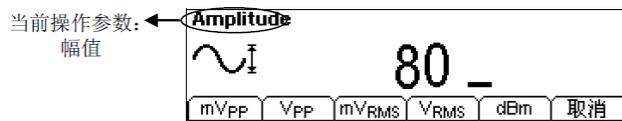


图 2-3 设置幅值的参数值

提示说明

幅值设置中的“dBm”单位选项只有在输出阻抗设置为 50Ω 时才会出现。

设置偏移电压

- 1、按 Sine→偏移/低电平→偏移，设置偏移电压参数值。

屏幕显示的偏移电压为上电时的默认值，或者是预先选定的偏移量。在更改参数时，如果当前偏移量对于新波形是有效的，则继续使用当前偏移值。

- 2、输入所需的偏移电压。

使用数字键盘或旋钮，输入所选参数值，然后选择偏移量所需单位，按下对应于所需单位的软键。



图 2-4 设置偏移量的参数值

设置起始相位

- 1、按 Sine→相位，设置起始相位参数值。

屏幕显示的初始相位为上电时的默认值，或者是预先选定的相位。在更改参数时，如果

当前相位对于新波形是有效的，则继续使用当前偏移值。

2、输入所需的相位。

使用数字键盘或旋钮，输入所选参数值，然后选择单位。

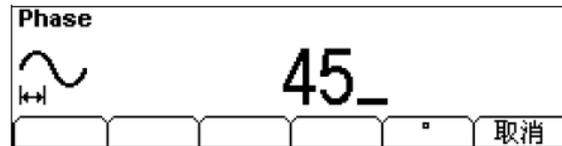


图 2-5 设置相位参数值

此时按 View 键切换为图形显示模式，查看波形参数，如下图所示。

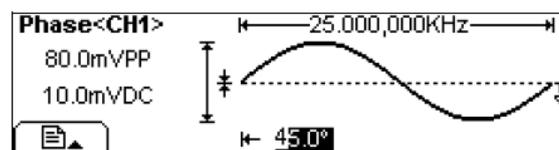


图 2-6 图形显示模式下的波形参数

输出单通道波形

输出正弦波

如何输出一个频率为 20kHz，幅值为 2.5Vpp，偏移量为 500mVDC，初始相位为 10°的正弦波形。

操作步骤：

1、设置频率值

按 Sine→按频率/周期软键切换，软键菜单频率反色显示；

使用数字键盘输入“20”，选择单位“kHz”，设置频率为 20kHz。

2、设置幅度值

按幅值/高电平软键切换，软键菜单幅值反色显示。

使用数字键盘输入“2.5”，选择单位“Vpp”，设置幅值为 2.5Vpp。

3、设置偏移量

按偏移/低电平软键切换，软键菜单偏移反色显示。

使用数字键盘输入“500”，选择单位“mVDC”，设置偏移量为 500mVDC。

4、设置相位

按相位软键使其反色显示。

使用数字键盘输入“10”，选择单位“°”，设置初始相位为10°。

上述设置完成后，按 View 键切换为图形显示模式，信号发生器输出下图所示正弦波。

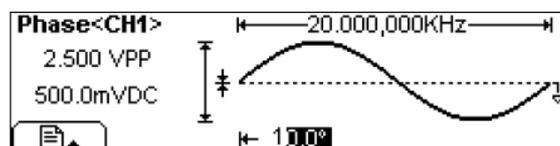


图 3-1 输出正弦波形

输出脉冲波

如何输出一个频率为 5kHz，幅值为 50mVpp，偏移量为 5mVDC，脉宽为 20 μs(占空比为 10%)，延时为 200 μs 的脉冲波形。

操作步骤：

1、设置频率值

按 Pulse→按频率/周期软键切换，软键菜单频率反色显示；

使用数字键盘输入“5”，选择单位“kHz”，设置频率为 5kHz。

2、设置幅度值

按幅值/高电平软键切换，软键菜单幅值反色显示；

使用数字键盘输入“50”，选择单位“mVpp”，设置幅值为 50mVpp。

3、设置偏移量

按偏移/低电平软键切换，软键菜单偏移反色显示；

使用数字键盘输入“5”，选择单位“mVDC”，设置偏移量为 5mVDC。

4、设置脉宽（占空比）

按脉宽（占空比），软键菜单脉宽（占空比）反色显示；

使用数字键盘输入“20”（“10”），选择单位“μs”（“%”），设置脉宽为 20 μs（占空比为 10%）。

5、设置延时

按延时，软键菜单延时反色显示；

使用数字键盘输入“200”，选择单位“μs”，设置延时为 200 μs。

上述设置完成后，按 View 键切换为图形显示模式，信号发生器输出下图所示脉冲波

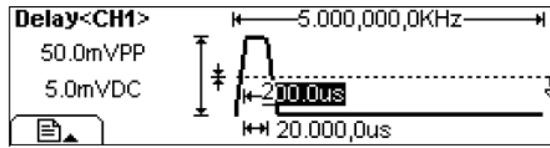


图 3-4 输出脉冲波形

输出噪声波

如何输出一个幅值为 30mVpp，偏移量为 10mVDC 的噪声波形。

操作步骤：

1、设置幅度值

按 Noise→幅值/高电平软键切换，软键菜单幅值反色显示；

使用数字键盘输入“30”，选择单位“mVpp”，设置幅值为 30mVpp。

2、设置偏移量

按偏移/低电平软键切换，软键菜单偏移反色显示；

使用数字键盘输入“10”，选择单位“mVDC”，设置偏移量为 10mVDC。

上述设置完成后，按 View 键切换为图形显示模式，信号发生器输出下图所示噪声波。



图 3-5 输出噪声波形