

# 自动控制理论 实验指导书

同济大学 汽车学院

2013 年 9 月

# 目录

<b>一MATLAB/ Simulink及RTW软件介绍 .....</b>	<b>3</b>
1. Simulink仿真集成环境简介 .....	3
2. RTW介绍 .....	11
<b>三 控制理论基础实验 .....</b>	<b>14</b>
一、电子节气门介绍 .....	14
二、电子节气门建模 .....	16
实验一 稳态误差及稳定性分析 .....	19
实验二 线性系统串联校正 .....	23
实验三 电子节气门控制器设计 .....	27

## 一 MATLAB/ Simulink 及 RTW 软件介绍

### 1. Simulink 仿真集成环境简介

Simulink 是可视化动态系统仿真环境。1990 年正式由 Mathworks 公司引入到 MATLAB 中，它是 Simulation 和 Link 的结合。这里主要介绍它的使用方法和它在控制系统仿真分析和设计操作的有关内容。

#### 1、进入 Simulink 操作环境

双击桌面上的MATLAB图标，启动MATLAB，进入开发环境，如图 1所示：

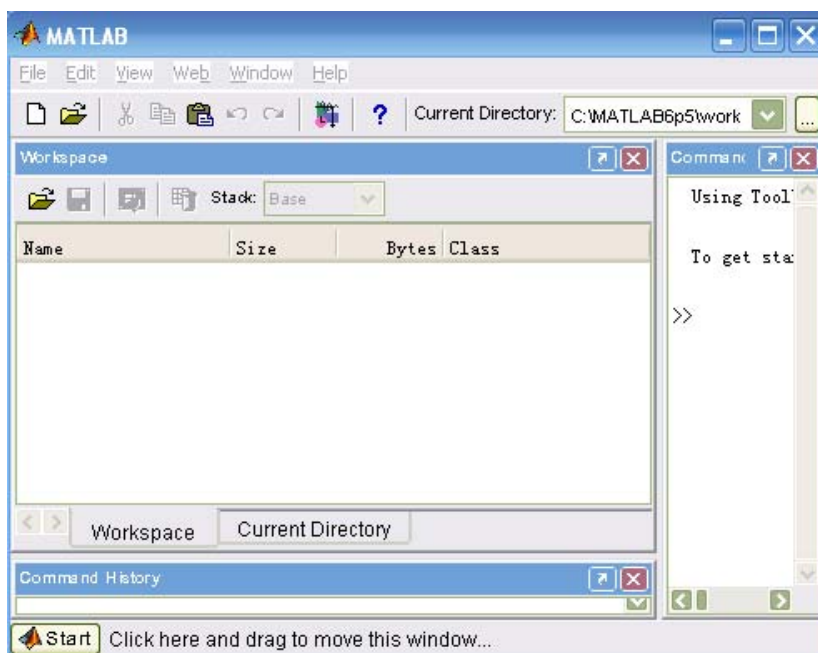





图 1 MATLAB 开发环境

从 MATLAB 的桌面操作环境画面进入 Simulink 操作环境有多种方法，介绍如下：

- ①点击工具栏的Simulink图标，弹出如图 2的图形库浏览器画面。
- ②在命令窗口键入“simulink”命令，可自动弹出图形库浏览器。

上述两种方法需从该画面“File”下拉式菜单中选择“New/Model”，或点击图标，得到图 3的图形仿真操作画面。

- ③从“File”下拉式菜单中选择“New/Model”，弹出图 3所示的未命名的图形仿真操作画面。从工具栏中点击图形库浏览器图标，调出图 2的图形库浏览器画面。图 3用于仿真操作，图 2的图形库用于提取仿真所需的功能模块。

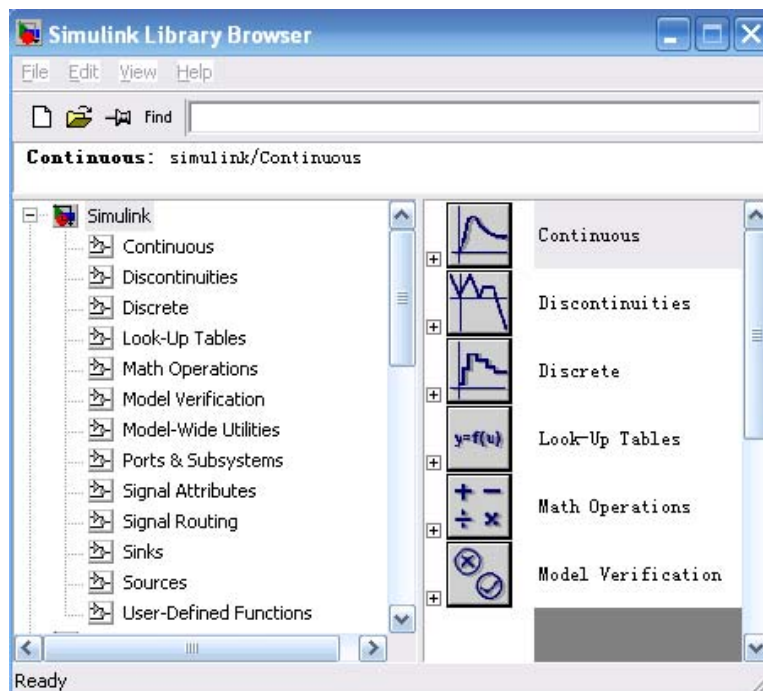


图 2 Simulink 图形库浏览器画面

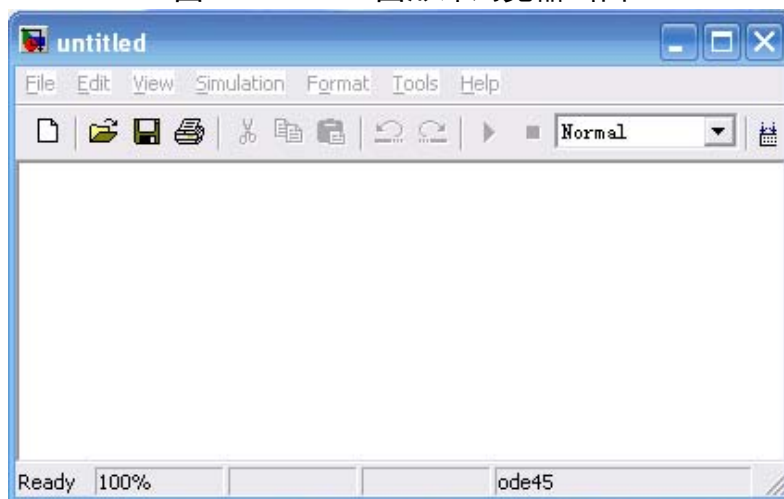


图 3 Simulink 仿真操作环境画面

## 2、提取所需的仿真模块

在提取所需仿真模块前，应绘制仿真系统框图，并确定仿真所用的参数。图 2 中的仿真用图形库，提供了所需的基本功能模块，能满足系统仿真的需要。该图形库有多种图形子库，用于配合有关的工具箱。下面将对本书中实验可能用到的功能模块作一个简单介绍。

### (1) Sources（信号源模块组）

点击图 2 图形库浏览器画面中的 Sources，界面右侧会出现各种常用的输入信号，如图 4 所示。

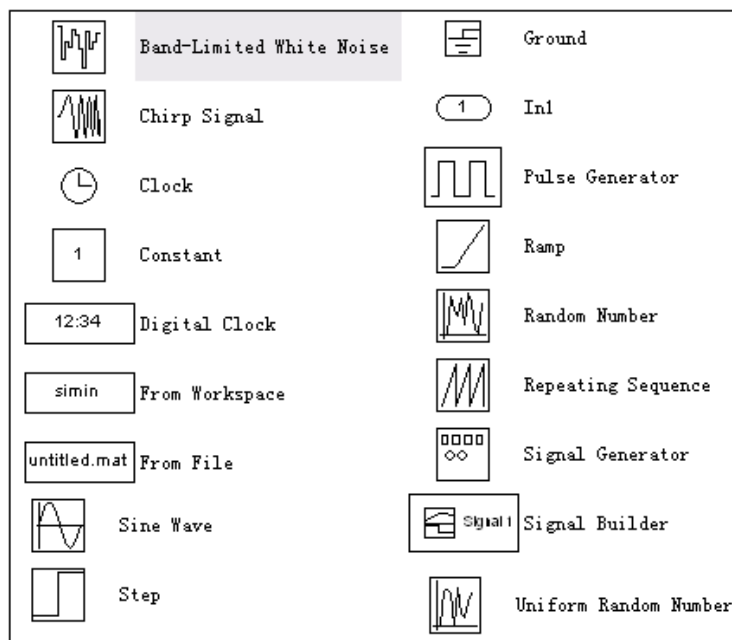


图 4 信号源模块组

- **In**（输入端口模块）——用来反映整个系统的输入端子，这样的设置在模型线性化与命令行仿真时是必需的。
- **Signal Generator**（信号源发生器）——能够生成若干种常用信号，如方波信号、正弦波信号、锯齿波信号等，允许用户自由调整其幅值、相位及其它信号。
- **From File**（读文件模块）和 **From Workspace**（读工作空间模块）——两个模块允许从文件或 MATLAB 工作空间中读取信号作为输入信号。
- **Clock**（时间信号模块）——生成当前仿真时钟，在于事件有关的指标求取中是很有意义的。
- **Constant**（常数输入模块）——此模块以常数作为输入，可以在很多模型中使用该模块。
- **Step**（阶跃输入模块）——以阶跃信号作为输入，其幅值可以自由调整。
- **Ramp**（斜坡输入模块）——以斜坡信号作为输入，其斜率可以自由调整。
- **Sine Wave**（正弦信号输入模块）——以正弦信号作为输入，其幅值、频率和初相位可以自由调整。
- **Pulse Generator**（脉冲输入模块）——以脉冲信号作为输入，其幅值和脉宽可以自由调整。

## （2）Continuous（连续模块组）

连续模块组包括常用的连续模块，如图 5所示。

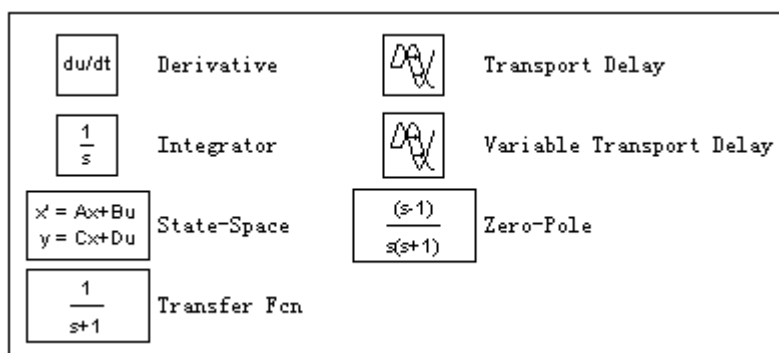


图 5 连续模块组

- **Derivative**（微分器）——此模块相当于自动控制系统中的微分环节，将其输入端的信号经过一阶数值微分，在其输出端输出。在实际应用中应该尽量避免使用该模块。

- **Integrator**（积分器）——此模块相当于自动控制系统中的积分环节，将输入端信号经过数值积分，在输出端输出。

- **Transfer Fcn**（传递函数）——此模块可以直接设置系统的传递函数，以多项式的比值形式描述系统，一般形式为  $G(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0}$ ，其分子分母多项式的系数可以自行设置。

- **Pole-Zero**（零极点）——将传递函数分子和分母分别进行因式分解，变成零极点表达式  $G(s) = K \frac{(s - z_1)(s - z_2) \dots (s - z_m)}{(s - p_1)(s - p_2) \dots (s - p_n)}$ ，其中  $z_i$ （系统的零点）、 $p_j$ （系统的极点）可以自行设置。

- **Transport Delay**（时间延迟）——此模块相当于自动控制系统中的延迟环节，用于将输入信号延迟一定时间后输出，延迟时间可以自行调整。

### （3）Math Operations（数学函数模块组）

数学函数模块组包含各种数学函数运算模块，如图 6所示。

- **Gain**（增益函数）——此模块相当于自动控制系统中的比例环节，输出信号等于输入信号的乘以模块中指定的数值，此数值可以自行调整。

- **Sum**（求和模块）——此模块相当于自动控制系统中的加法器，将输入的多路信号进行求和或求差。

- 其它数学函数，如 **Abs**（绝对值函数）、**Sign**（符号函数）、**Rounding Function**（取整

模块) 等。

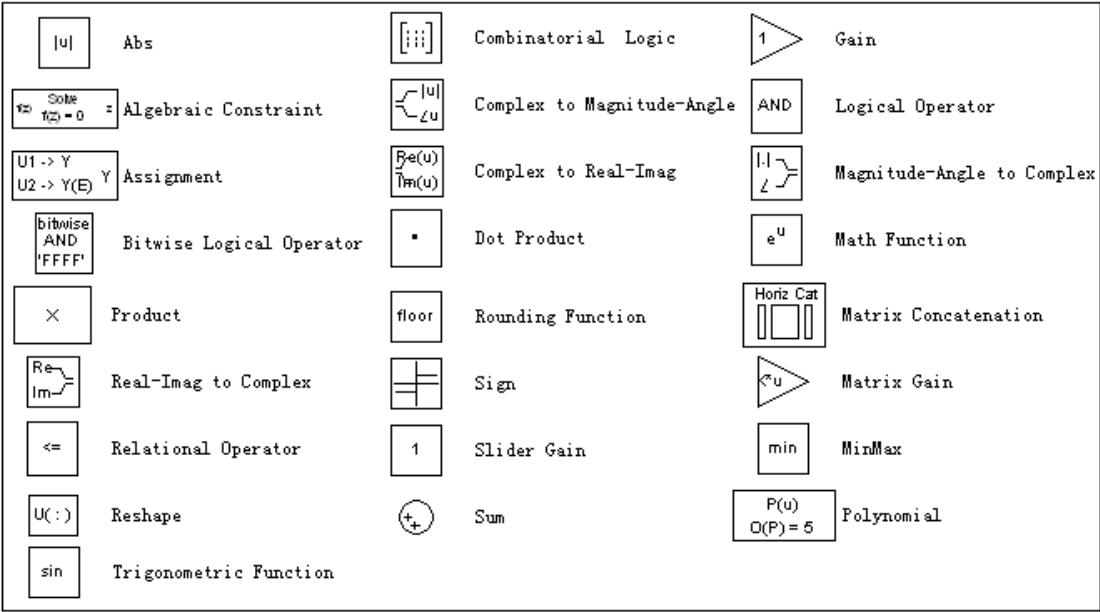


图 6 数学函数模块组

(4) Sinks (输出池模块组)

输出池模块组包含那些能显示计算结果的模块，如图 7所示。

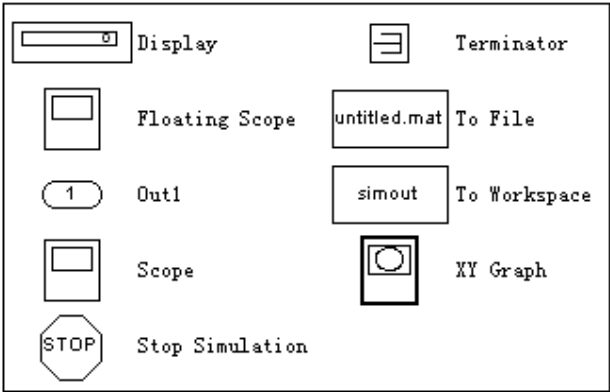


图 7 输出池模块组

- **Out**（输出端口模块）——用来反映整个系统的输出端子，这样的设置在模型线性化与命令行仿真时是必需的，另外，系统直接仿真时这样的输出将自动在 **MATLAB** 工作空间中生成变量。
- **Scope**（示波器模块）——将其输入信号在示波器中显示出来。
- **x-y Graph**（x-y 示波器）——将两路输入信号分别作为示波器的两个坐标轴，将信号的相轨迹显示出来。
- **To Workspace**（工作空间写入模块）——将输入的信号直接写到 **MATLAB** 的工作空

间中。

- **To File**（写文件模块）——将输入的信号写到文件中。
- **Display**（数字显示模块）——将输入的信号以数字的形式显示出来。
- **Stop Simulation**（仿真终止模块）——如果输入的信号为非零时，将强行终止正在进行的仿真过程。
- **Terminator**（信号终结模块）——可以将该模块连接到闲置的未连接的模块输出信号上，避免出现警告。

从图 2 中用鼠标点击打开所需子图形库，用鼠标选中所需功能模块，将其拖曳到图 3 中的空白位置，重复上述拖曳过程，直到将所需的全部功能模块拖曳到图 3 中。

拖曳时应注意下列事项：

- （1）根据仿真系统框图，选择合适的功能模块进行拖曳，放到合适的位置，以便于连接。
- （2）对重复的模块，可采用复制和粘贴操作，也可以反复拖曳。
- （3）功能模块和图 3 的大小可以用鼠标移动到图标或图边，在出现双向箭头后进行放大或缩小的操作。
- （4）选中功能模块的方法是直接点击模块，用鼠标选定所需功能模块区域来选中区域内所有功能模块和连接线，点击选中，并按下“shift”键，再点击其它功能模块。

### 3、功能模块的连接

根据仿真系统框图，用鼠标点击并移动所需功能模块到合适的位置，将鼠标移到有关功能模块的输出端，选中该输出端并移动鼠标到另一个功能模块的输入端，移动时出现虚线，到达所需输入端时，释放鼠标左键，相应的连接线出现，表示该连接已完成。重复以上的连接过程，直到完成全部连接，组成仿真系统。

### 4、功能模块参数设置

使用者需设置功能模块参数后，方可进行仿真操作。不同功能模块的参数是不同的，用鼠标双击该功能模块自动弹出相应的参数设置对话框。

例如，图 8 是 **Transfer Fcn**（传递函数）功能模块的对话框。功能模块对话框由功能模块说明和参数设置框组成。功能模块说明框用于说明该功能模块使用的方法和功能，



参数框用于设置该模块的参数。Transfer Fcn的参数框由分子和分母多项式两个编辑框组成，在分子多项式框中，用户可输入系统模型的分子多项式，在分母多项式框中，输入系统模型的分母多项式。设置功能模块的参数后，点击OK进行确认，将设置的参数送仿真操作画面，并关闭对话框。

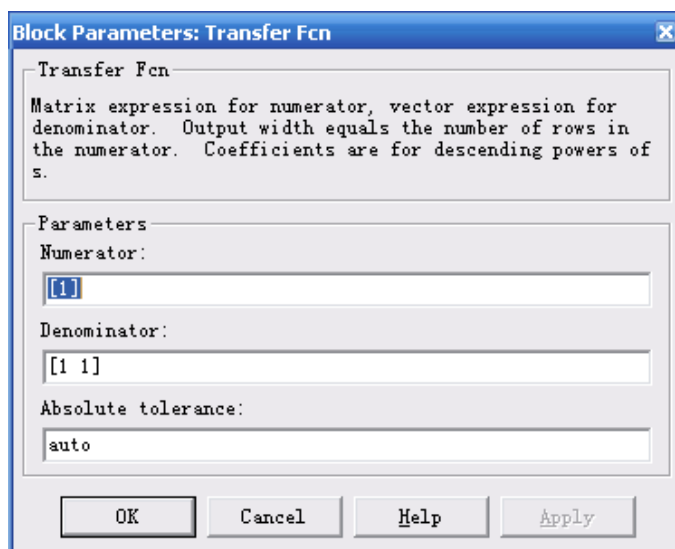





图 8 Transfer Fcn（传递函数）模块参数设置对话框

## 5、仿真器参数设置

点击图 3操作画面“Simulation”下拉式菜单“Simulation Parameters…”选项，弹出如图 9所示的仿真参数设置画面。共有Solver、Workspace I/O、Diagnostics、Advanced和Real-Time Workshop等五个页面。在Solver中设置Solver Type、Solver（步长）等。仿真操作时，可根据仿真曲线设置终止时间和最大步长，以便得到较光滑的输出曲线。

## 6、示波器参数设置

当采用示波器显示仿真曲线时，需对示波器参数进行设置。双击 Scope 模块，弹出如图 10 所示的示波器显示画面，点击画面的图标，弹出如图 11 所示的示波器属性对话框，分 2 个页面，用于设置显示坐标窗口数、显示时间范围、标记和显示频率或采样时间等。时间范围可以在示波器属性对话框里的 General 页中的 Time range 设置，设置值应与仿真器终止时间一致，以便最大限度显示仿真操作数据。鼠标右键点击示波器显示窗口，从弹出菜单选择“Autoscale”，或直接点击图标，可在响应曲线显示后自动调整纵坐标范围；从弹出菜单选择“Save current axes settings”，或直接点击图标，

将当前坐标轴范围的设置数据存储。此外，还有打印、放大或恢复等操作。

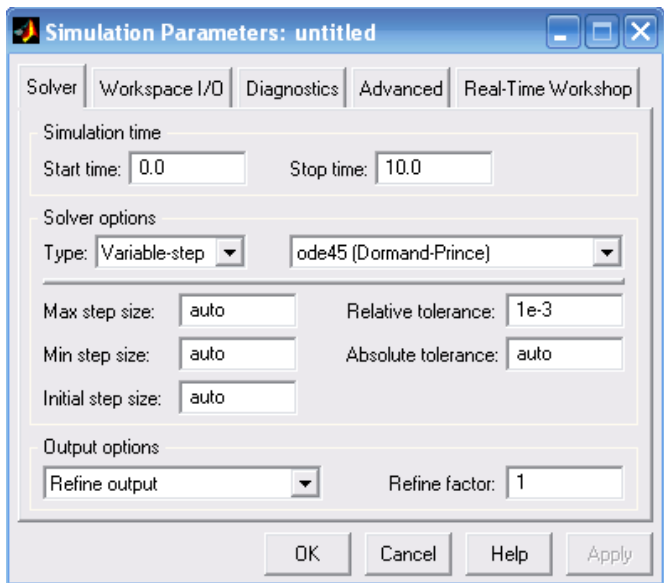


图 9 仿真参数设置画面

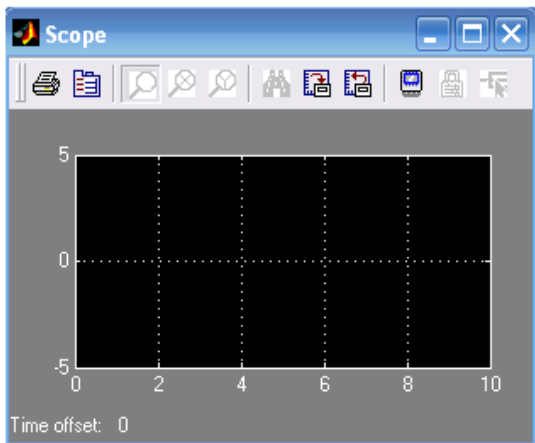


图 10 示波器显示画面

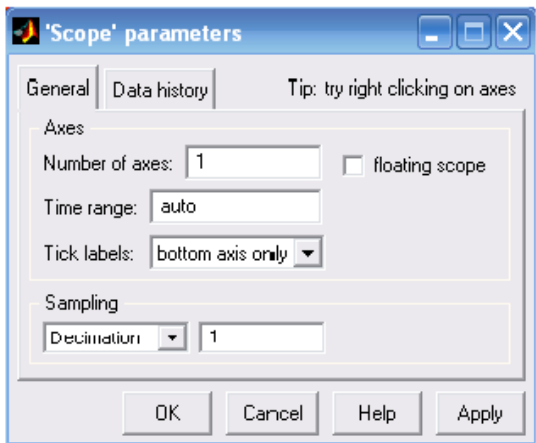


图 11 示波器属性对话框

7、运行仿真

模型编辑好后，点击图 3操作画面 “Simulation” 下拉式菜单 “start” 或 “start Simulation” 按钮运行，双击Scope模块，显示输出曲线。

8、对数据作后续处理

当仿真任务比较复杂时，需要将 Simulation 生成的数再导入到工作空间进行分析和处理，仿真结束后，输出结果通过 “To workspace” 传送到工作空间中，在工作空间窗口中能看到这些变量，使用 “whos” 命令能看到这些变量的详细信息。另外，“To file”、“From file” 模块能实现文件与 Simulink 的数据传输。

## 2. RTW 介绍

RTW(Real-Time Windows Target)是 MATLAB 图形建模和仿真环境 Simulink 的一个重要的补充功能模块, 是一个基于 Simulink 的代码生成环境。以往我们在设计控制系统中, 先在 Simulink 中建模和仿真, 直至仿真结果满意为止。但是真正要开发实际控制器时, 我们需要重新使用 C 或者其它语言重新编程, 而当设计好的控制器与实际被控系统相联后, 可能控制效果就不如仿真效果那么理想, 因为在仿真时我们可能对模型作了一些假设和近似或者忽略了实际系统可能受到的扰动和噪声, 这样造成仿真与实际应用相脱节, RTW 的产生解决了这个问题。

RTW 支持两种类型的实时目标设计: 一种是快速原型化目标设计, 另一种是嵌入式目标设计。所谓快速原型化目标设计又叫半实物仿真, 就是指可以用 Simulink 设计出来的控制器直接去控制实际的被控对象, 通过半实物仿真过程来观察控制效果, 如果控制效果不理想, 则可以直接在 Simulink 上调整控制器的结构或参数, 直至获得满意控制结果。这样调试好的控制器可以认为是实际控制器的原型 (Prototype), 显然此时的控制器显然要好于纯数字仿真下的控制器。嵌入式目标设计就是将 Simulink 下调试好的控制器直接生成 C 语言程序, 并经过编译、连接生成可执行的应用程序, 下载并嵌入到控制计算机上, 使控制计算机可以脱离 MATLAB/Simulink 环境直接用于实时控制。RTW 工具箱提供了多个目标环境设置, 例如有通用实时目标 (GRT)、通用实时 Malloc 目标、Tornado 目标、DOS 目标等捆绑目标和实时目标 (RTW)、XPC 等独立目标。利用这些目标, 用户能快速, 高质量的完成系统原型化设计和实时系统构建。

### 1、RTW 的功能

- 1) Simulink 代码生成器: 能自动的从 Simulink 模型中产生 C 和 Ada 代码。
- 2) 创建过程: 可扩展的程序创建过程使用户产生自己的产品级或快速原型化目标
- 3) Simulink 外部模式: 外部模式使 Simulink 与运行在实时测试环境下的模型之间或在相同计算机上的另一个进程进行通信成为可能。外部模式使用户将 Simulink 作为前向终端进行实时的参数调整或数据观察。
- 4) 多目标支持: 使用 RTW 捆绑的目标, 用户可以针对多种环境创建程序,

包括 Tomdao 和 DOS 环境。通用实时目标和嵌入式实时目标为开发个性化的快速原型环境或产品目标环境提供了框架。除了捆绑的目标外，实时视窗目标或 xPC 目标(两者都为独立产品)使用户可以将任何形式的 PC 机变成一个快速原型化目标，或者中小容量的产品级目标。

5) 快速仿真:使用 Simulink 加速器、S 函数目标或快速仿真目标，用户能以平均 5~20 倍的速度加速仿真过程。

## 2、RTW 的应用

- 1) 产品级的嵌入式实时应用领域
- 2) 快速原型化：图形化/性能评估/完善设计/在线调试
- 3) 实时仿真
- 4) 生成完善的实时解决途径：支持多种类型的控制及数字处理应用
- 5) 知识产权保护
- 6) 快速仿真：Simulink 加速器/快速仿真目标/S 函数目标

## 3、RTW 支持的目标环境

- 1) 快速仿真目标 (Rapid Simulation Target)
  - 2) S 函数目标和加速器目标 (S-Function and Acceleration Target)
  - 3) 嵌入式目标 (Embedded Target)
  - 4) 快速原型化目标
  - 5) 完善的快速原型化解决途径 (Turnkey Rapid Prototyping Target)
- 实时视窗目标 (Real-Time Windows Target)、xPC 目标(xPC Target)

## 4、RTW 支持的编译器

Borland、LCC、Microsoft Visual C/C++、Watcom，可以通过在 MATLAB 命令窗运行 `mex -setup` 选择

## 5、RTW 操作步骤

1)C 编译器的安装：在 Matlab 命令栏输入：`mex -setup`，根据步骤提示进行安装。

2) Kernel 的安装：在 Matlab 命令栏输入：`rtwintgt -install`，根据步骤提示进行安装。

3) 创建 Simulink 模型，并进行仿真参数的设置，Solver 选项卡设置 Type

为 Fixed-step, 并推荐选择 ode5 算法; Real-Time Workshop 选项卡设置 Target selection 选项, 单击 Browse 按钮打开系统目标文件浏览器, 选定 Real-time windows target 选项, 并单击 OK 按钮, 其它参数为默认值即可。

4) 确保软硬件都连接好之后, 开始仿真运行。在 Simulink 的 tool 菜单下选择 real-time workshop 下的 build model; 在 Simulink 中选中 external, 点击 connect to target, 在 Matlab 命令栏中显示: Model rtwin\_model loaded; 在菜单栏 Simulink 中选择 start real time code, 仿真终止以后在 Matlab 命令栏中显示: Model rtwin\_model unloaded。

### 三 控制理论基础实验

#### 一、电子节气门介绍

节气门是控制空气进入发动机的一道可控阀门，气体进入进气管后会和汽油混合成可燃混合气，从而燃烧做功。它上接空气滤清器，下接发动机缸体，被称为是汽车发动机的咽喉。节气门有传统拉线式和电子节气门两种，传统发动机节气门操纵机构是通过拉索（软钢丝）或者拉杆，一端连接油门踏板，另一端连接节气门的连动板而工作，节气门由脚踏板带动油门拉线控制，这种拉线控制的节气门在急加速等特殊工况时易出现进气迟滞现象，而且节气门处在气流扰动下不平稳，因此空气量无法精确控制，加速不理想且不稳定。电子节气门主要通过节气门位置传感器，来根据发动机所需能量，控制节气门的开启角度，从而调节进气量的大小，电子节气门可根据节气门位置信号，ECU 直接驱动直流电动机快速地做出响应，及时将节气门打开所需的开度，而且电子节气门在自身减速机构的自锁作用下，不会因为气流的扰动而波动，以保证发动机的进气量和转速的稳定。

电子节气门的研究工作起源于 20 世纪 70 年代，80 年代开始有产品问世，这些年来，国外对电子节气门的研究取得了非常迅速的发展。发展趋势可总结为：在控制策略上由线性控制发展为非线性控制，由辅助电子节气门发展为独立的电子节气门系统，从单一的控制功能发展到集成多种控制功能，兼顾提高动力性、经济性、操纵稳定性、排放性和乘坐舒适性。

国外多家公司已对电子节气门的系统作了深入的研发，比如德国Bosch，Pierburg，美国Delphi，Visteon，日本Toyota，Hitachi，Denso，意大利Marelli等已推出系列化产品应用于各种品牌的中高档轿车。虽然国内某些轿车，如POLO，也配备了电子节气门系统，但当前对ETC还没有系统深入的研究，也没有成熟的产品。当前，虽然国内部分高级轿车，如宝来、奥迪、帕萨特、POLO、红旗等已经配备了电子节气门控制系统，但都属于国外引进的技术，对其核心技术了解得很少。可喜的是，这些年来，中国第一汽车集团公司开发了电子节气门控制系统，并把该项技术用于红旗HQ3 高级轿车上。

电子节气门系统由驱动电机，减速齿轮组，节气门，复位弹簧，位置传感器等部件组成，其内部结构如图 12所示。其中电动步进电机由来自电机驱动的

信号控制，一个同心的弹簧控制节气门，当节气门打开到接近关闭的位置，如果步进电机电源丢失，弹簧将关闭节气门到限位位置，当节气门完全关闭的时候，弹簧也将尽力打开节气门，节气门开启是利用节气门上的直流电机通过减速机构来自动实现的。

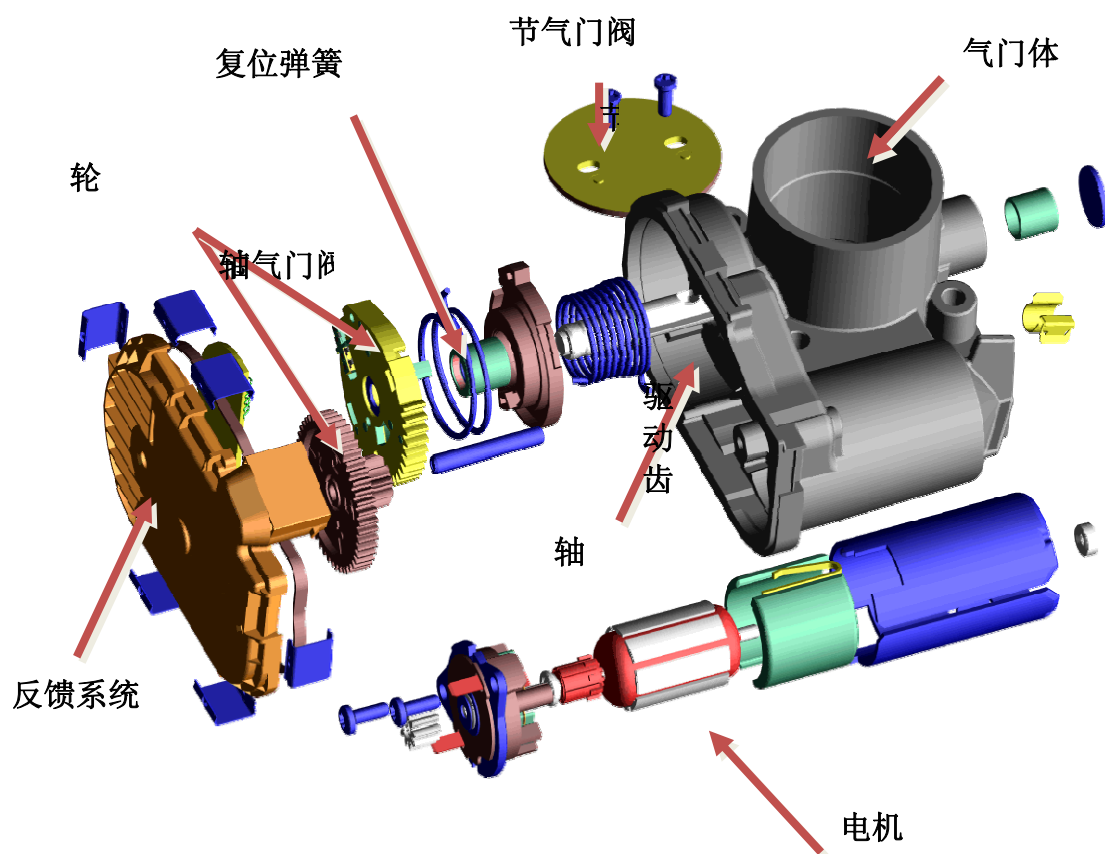


图 12 电子节气门内部构造

## 二、电子节气门建模

电子节气门控制系统由控制器，电机驱动器及节气门本体组成。其中节气门本体又包括：驱动电机，减速齿轮组，阀门，复位弹簧，位置传感器等部分，结构如图 13所示。

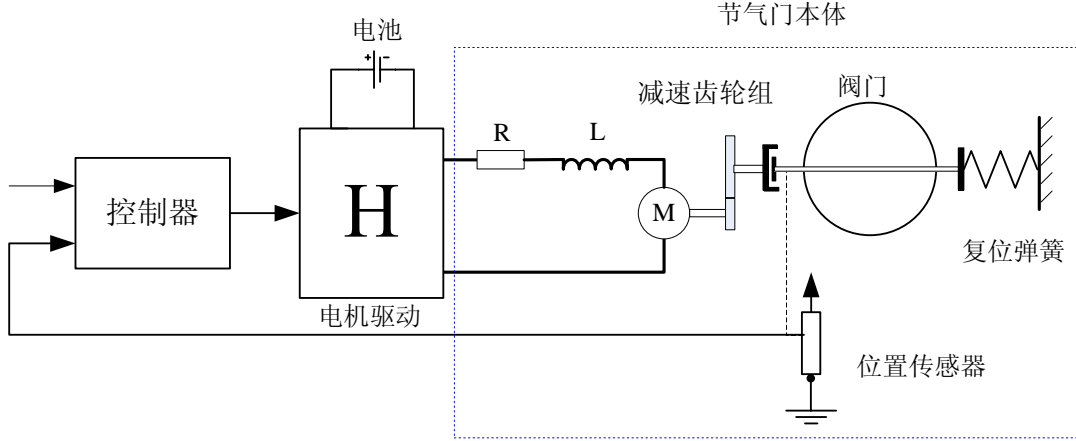


图 13 电子节气门控制系统结构图

节气门本体模型构建如下：

根据基尔霍夫定律，节气门电机的电枢电路方程为：

$$R_a i_a + L_a \cdot \frac{di_a}{dt} + V_b = E_a \quad (1)$$

$$V_b = k_b \dot{\theta}_m \quad (2)$$

$$i_a = \frac{T_m}{k_t} \quad (3)$$

其中， $k_b$ 为电机反电动势常数， $V_b$ 为电机反电动势， $L_a$ 为电机电感， $R_a$ 为电机电阻， $i_a$ 为电机电流， $E_a$ 为电机输入电压， $\theta_m$ 为电机旋转角， $T_m$ 为电机转矩， $k_t$ 为电机扭矩常数，把式(2)代入式(3)，得：

$$L_a \cdot \frac{di_a}{dt} = -R_a i_a - k_b \dot{\theta}_m + E_a \quad (4)$$

考虑 $L_a$ 非常非常小，忽略电枢电流的动态特性，即令 $L_a \cdot \frac{di_a}{dt} = 0$ ，得：



$$\dot{i}_a = \frac{E_a}{R_a} - \frac{k_b}{R_a} \dot{\theta}_m \quad (5)$$

根据扭矩守恒原理，电机旋转角  $\theta_m$  和节气门旋转角  $\theta$  的动力学方程为：

$$J_m \ddot{\theta}_m = -k_m \dot{\theta}_m - T_L + T_m \quad (6)$$

$$J_g \ddot{\theta} = -T_{sp} - T_{tf} - T_d + T_g \quad (7)$$

其中： $J_m$  和  $J_g$  分别为电机和节气门挡板转轴的转动惯量， $k_m$  为电机轴阻尼系数。

上述扭矩平衡方程中， $T_{tf}$  和  $T_{sp}$  分别为节气门摩擦力扭矩和节气门复位弹簧扭矩，由下两式计算（忽略非线性，并假设默认开度为 0 度）：

$$T_{tf} = k_f \dot{\theta} \quad (8)$$

$$T_{sp} = k_{sp} \theta \quad (9)$$

其中： $k_r$  为滑动摩擦力系数， $k_{tf}$  为静态摩擦力系数， $k_{sp}$  为弹簧弹性系数， $T_d$  为空气流量负载扭矩，考虑为未知扰动。另外， $T_L$  和  $T_g$  分别为负载扭矩和齿轮传动比扭矩，考虑减速齿轮组的理想传动比关系  $n = \frac{\theta}{\theta_m} = \frac{T_g}{T_L}$ ，则由式(6)、(7)代入(3)可得：

$$T_g = -n^2 J_m \ddot{\theta}_m - n^2 k_m \dot{\theta}_m + n k_t i_a \quad (10)$$

另外，由式(5)可知上式中的  $i_a$  可表达为：

$$i_a = \frac{E_a}{R_a} - \frac{n k_b}{R_a} \dot{\theta} \quad (11)$$

将(10)和(11)代入(7)，考虑到(8)、(9)，整理得：

$$\ddot{\theta} + a \dot{\theta} + b \theta = c \left( u + \frac{1}{c} d \right) \quad (12)$$

为方便设计与分析控制系统，将未建模动态和未知干扰等不确定性合并，描述为叠加干扰  $d = \frac{T_d}{J}$ ，其中， $a = \frac{(n^2 k_m + k_f + \frac{n^2 k_b k_t}{R_a})}{J}$ ， $J = n^2 J_m + J_g$ ， $b = \frac{k_{sp}}{J}$ ，

$c = \frac{n k_t}{J R_a}$   $u \in [-E_a, E_a]$  为控制输入电压。以上物理参量具体值可参考表 1。

表 1 模型参数名义值

$k_b / (\frac{V}{rad/s})$	$k_f / (\frac{N \cdot m}{rad/s})$	$k_m / (\frac{N \cdot m}{rad/s})$	$k_{sp} / (\frac{N \cdot m}{rad})$	$k_{tf} / (N \cdot m)$	$R_a / \Omega$
0.016	$4.0 \times 10^{-4}$	$1.6 \times 10^{-6}$	0.0247	0.0048	2.8
$\theta_0 / rad$	$J / kg \cdot m^2$	$k_t / (N \cdot m \cdot A^{-1})$	$k_{pre} / (N \cdot m)$	$E_a / V$	
0.0349	$1.15 \times 10^{-3}$	0.016	0.107	12	
$a$	$b$	$c$	$d$		
21	23	84	17		

最终可以得到系统结构如下图 14所示

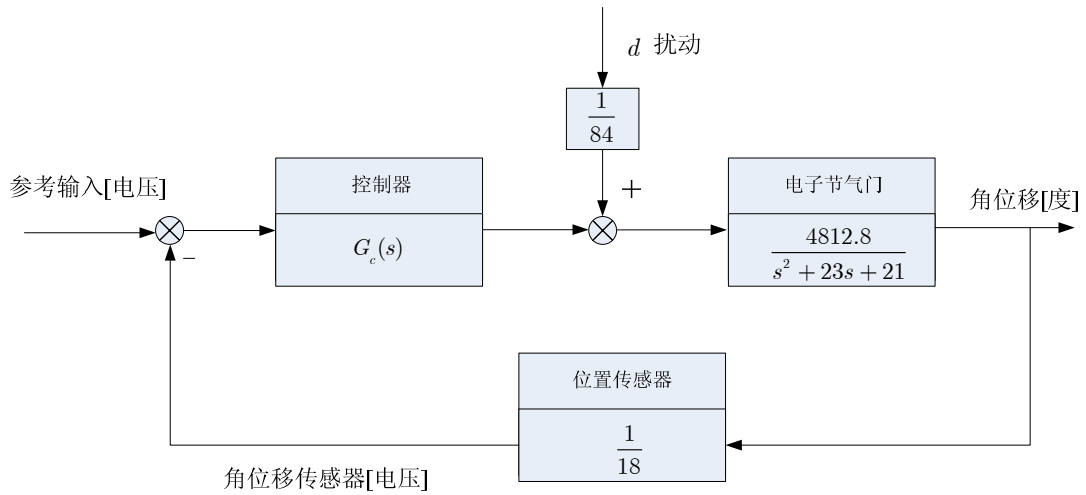


图 14 电子节气门系统传递函数

其中，电子节气门为二阶系统，控制器 $G_c(s)$ 为校正装置，位置传感器实现角位移检测，其传递函数为比例环节；系统的参考输入为电压，系统的输出为角位移。

## 实验一 稳态误差及稳定性分析

### 一、实验目的

1. 了解系统增益变化对系统稳定性以及稳态误差的影响；
2. 观察系统结构和稳态误差之间的关系。

### 二、实验内容以及实验报告要求：

请认真阅读指导书的第三部分**控制理论基础实验**中的（一电子节气门介绍；二电子节气门建模，这二部分内容）

#### 第一部分：

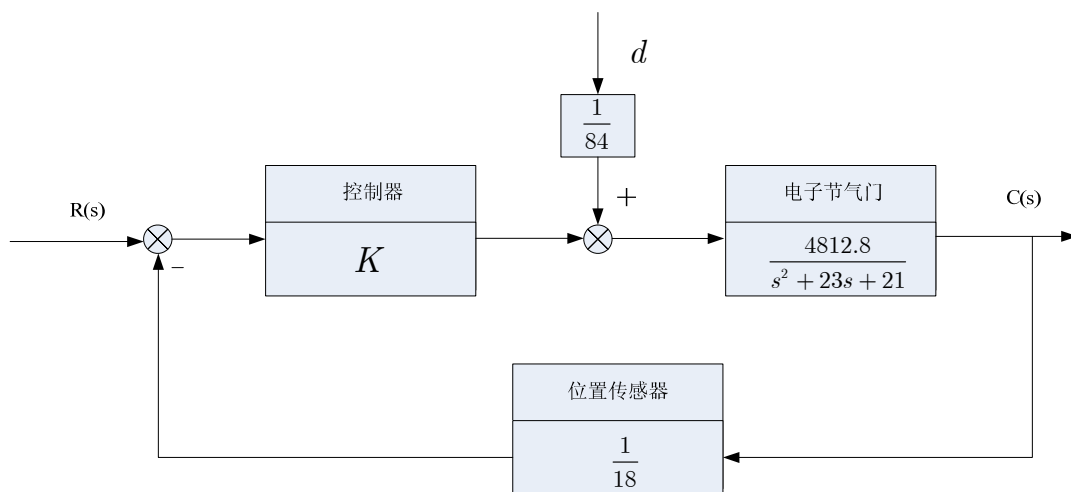


图 15 0 型系统

- 1) 建立如图 15所示的系统模型。
- 2) 根据劳斯判据计算  $K$  稳定的取值范围，取 3 个不同  $K=1,10,100$  值，在阶跃响应下观测并记录稳态误差，同时分别利用误差系数和误差传递函数的方法进行稳态误差计算并对比仿真结果。
- 3) 对系统添加气流扰动后，取原来 3 个不同  $K=1,10,100$  值，重新计算系统的稳态误差，并和仿真结果进行对比。
- 4) 将系统输入切换到斜坡输入，取  $K=1,10,100$ ，重新分别在受扰和不受扰情况下的系统稳态误差并和仿真结果对比。

第二部分：

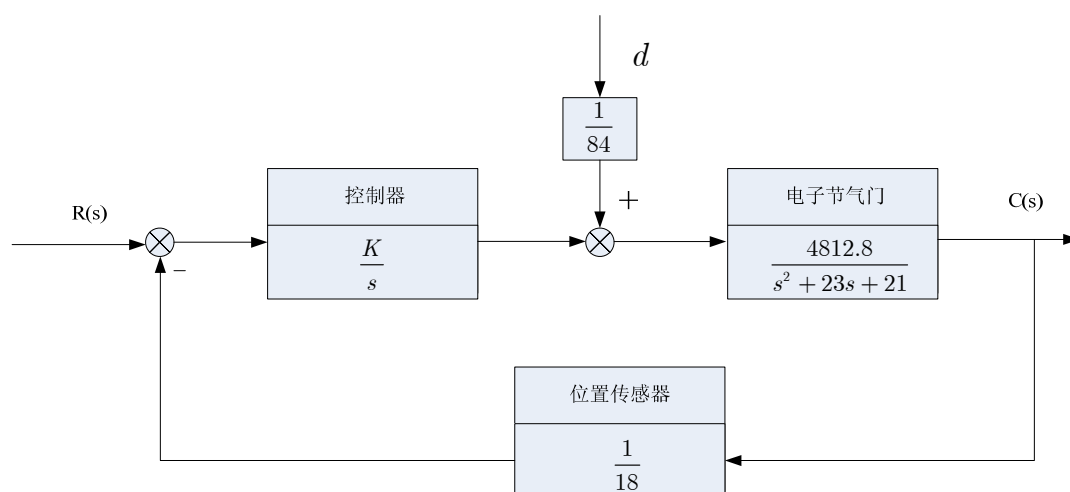


图 16 I 型系统

- 1) 建立如图 16所示的系统模型。
- 2) 根据劳斯判据计算  $K$  稳定的取值范围，取 3 个不同的  $K=0.01, 0.1, 1$  值，在阶跃响应下观测并记录稳态误差，同时分别利用误差系数和误差传递函数的方法进行稳态误差计算并对比仿真结果。
- 3) 对系统添加气流扰动后，取原来 3 个不同  $K=0.01, 0.1, 1$  值，重新计算系统的稳态误差，并和仿真结果进行对比。
- 4) 将系统输入切换到斜坡输入，取  $K=0.01, 0.1, 1$ ，计算系统稳态误差并和仿真结果对比。

三、实验报告模板

## 实验报告一： 稳态误差及稳定性分析

年级： 同组学生姓名：

指导老师： 实验日期： 成绩：

第一部分：

1) K 的取值范围：

2) 在阶跃输入和未受扰时系统稳态误差仿真及计算：

K 值	仿真的稳态误差	误差传递函数	计算所得稳态误差

3) 在阶跃输入和受扰时系统稳态误差仿真及计算：

K 值	仿真的稳态误差	误差传递函数	计算所得稳态误差

4) 在单位斜坡输入和未受扰时系统稳态误差仿真及计算：

K 值	仿真的稳态误差	误差传递函数	计算所得稳态误差

5) 在单位斜坡输入和受扰时系统稳态误差仿真及计算：

K 值	仿真的稳态误差	误差传递函数	计算所得稳态误差

第二部分：

1) K 的取值范围：

2) 在阶跃输入和未受扰时系统稳态误差仿真及计算：

K 值	仿真的稳态误差	误差传递函数	计算所得稳态误差

3) 在阶跃输入和受扰时系统稳态误差仿真及计算：

K 值	仿真的稳态误差	误差传递函数	计算所得稳态误差

4) 在单位斜坡输入和未受扰时系统稳态误差仿真及计算：

K 值	仿真的稳态误差	误差传递函数	计算所得稳态误差

5) 在单位斜坡输入和受扰时系统稳态误差仿真及计算：

K 值	仿真的稳态误差	误差传递函数	计算所得稳态误差

实验心得（可另附页）：

## 实验二 线性系统串联校正

### 一、实验目的

#### 2 了解校正必要性

1. 熟练绘制系统的 Nyquist 以及 Bode 图
3. 熟练掌握控制系统的串联校正方法

### 二、实验内容：

已知单位负反馈系统的开环传递函数为  $G(s) = \frac{1}{s(s+1)}$ ，试设计串联无源超前校正装置，使系统指标满足：

正装置，使系统指标满足：

- 1) 单位斜坡输入信号时稳态误差  $e_{ss} \leq 0.1 \text{ rad}$
- 2) 校正后系统的开环截止频率  $\omega_c \geq 4.4 \text{ rad/s}$  相位裕度  $\gamma \geq 45^\circ$ ，幅值裕度  $h \geq 10 \text{ dB}$

实验报告要求：

- 1) 利用 Matlab 绘制校正前、后阶跃响应曲线，Bode 图以及 Nyquist 曲线，并获取校正前、后相位裕度、幅值裕度、剪切频率、超调量、调节时间。
- 2) 根据实验，试说明校正前后系统的调节时间和超调量有何变化，相角裕度，穿越频率又有什么变化？
- 3) 试在同一张图中画出系统的单位斜坡输入及其系统输出。
- 4) 给出控制器设计过程。

### 三、实验报告模板

## 实验报告二：线性系统串联校正

年级：

同组学生姓名：

指导老师：

实验日期：

成绩：

1) 请在同一张图上用 **MATLAB** 绘制出校正前后的 **BODE** 图以及 **NYQUIST** 曲线（用不同颜色区分），并用在图上标注出相应的相位裕度、幅值裕度和剪切频率。

2) 请在同一张图上用 **MATLAB** 绘制出校正前后的单位斜坡响应曲线（用不同颜色区分），并用在图上标注出超调量和调节时间。



3) 试通过时域指标和频域指标相互关系解释校正前后相关指标的变化。

4) 请在同一张图上用 **MATLAB** 绘制出校正前后的响应曲线(用不同颜色区分), 并用在图上标注出稳态误差。

5) 给出控制器设计的过程

实验心得：

### 实验三 电子节气门控制器设计

#### 一、实验目的

1. 以电子节气门为例，了解汽车自动控制系统的组成、工作原理及其控制规律；
2. 学习 PID 控制器参数在实际系统中的整定方法；

#### 二、电子节气门实验平台介绍

本实验平台以 RTW 为基础将普通 PC 机模拟成宿主机和目标机，实现电子节气门 ETCS 的半实物实时仿真。就实时性而言，Windows 本身是一个非实时操作系统，一般而言其定时精度最多只能达到 1ms，因此对于一些高精度实时控制场合时不合适的。RTW 则应用一个小型实时内核，该内核以 PC 时钟作为基本时钟源并优先从 Windows 操作系统获得来自 PC 时钟的中断，这样就阻断了任何来自 Windows 操作系统的调用请求，从而保证了系统的实时性。一般而言，对于较小的模型可以保证 10kHz 以上的采样频率。

系统原理图如图 17 所示。图 17 中数据采集卡（PCI-6221）起着 A/D 和 D/A 的作用，它负责把模拟电压信号（节气门位置）经过采集后转换数字信号，同时又把驱动器的输入电压数字信号转换成模拟电压信号。图中板卡和 PC 机所交换的信号为数字信号，用后缀[D]表示，板卡和电机驱动系统以及电子节气门所交换信号为模拟信号，用后缀[A]表示。电子节气门采用 Bosch 公司的 DV-E5。电机驱动系统定制购买，其主要实现的功能如下：通过输入电压改变 PWM 占空比，从而控制电机的实时供电电压，其中输入电压 0-2.4V 正转，2.4V-2.6V 停转，2.6V-5.0V 反转。考虑到电机驱动时噪音问题，选取系统 PWM 载波频率为 12kHz。

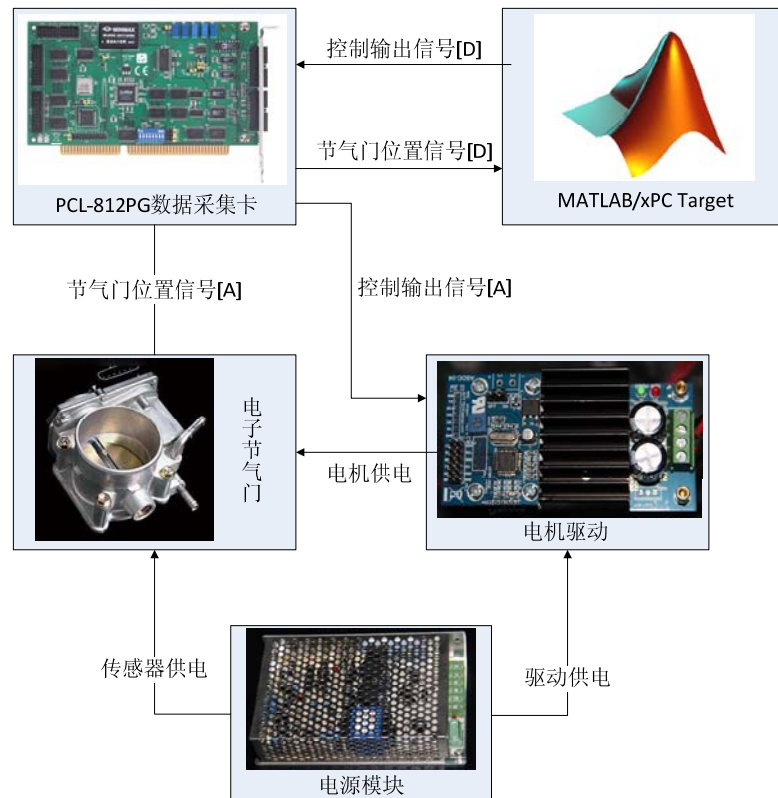


图 17 电子节气门实验样机系统原理图

### 三、实验内容及实验报告要求

- 1) 设定期望输入为 20 度到 80 度的阶跃，分别对 P, PI, PID 控制进行操作，并记录上升时间，超调量和稳态误差（ $\Delta = 2\%$ ）。
- 2) 在其它参数不变情况下，适当单独调节（增大或减小 20%）P, I, D 控制器参数（共 6 组），并记录其实际控制输出。
- 3) 简述 PID 控制器中,P,I,D 参数的作用，并用实验数据支撑。

### 四、实验报告模板

## 实验报告四：电子节气门控制器设计

年级: 同组学生姓名:

指导老师: \_\_\_\_\_ 实验日期: \_\_\_\_\_ 成绩: \_\_\_\_\_

1) 设定期望输入为 20 度到 80 度的阶跃, 分别对 P, PI, PID 控制进行操作, 并记录上升时间, 超调量和稳态误差。

控制器参数	上升时间	超调量	稳态误差 $\Delta = 2\%$

2) 在其它参数不变情况下, 适当单独调节 (增大或减小) P, I, D 控制器参数, 并记录其实际控制输出。

控制器参数	上升时间	超调量	稳态误差 $\Delta = 2\%$

3) 简述 PID 控制器中,PI,D 参数的作用,并用实验数据支撑。

4) 实验心得: