

# 通信系统仿真开发

——基于MATLAB、DSP及FPGA的设计与实现

沈良 任国春 高瞻 童晓兵 张玉明 编著



- 内容全面** 涵盖通信系统开发的经典内容和前沿技术
- 讲解新颖** 快速掌握业界主流的三大通信仿真软件开发
- 实用性强** 围绕通信系统要解决的实际问题展开讲解



机械工业出版社  
CHINA MACHINE PRESS

# 通信系统仿真开发

——基于 MATLAB、DSP 及 FPGA 的设计与实现

沈 良 任国春 高 瞻 童晓兵 张玉明 编著



机械工业出版社

本书在介绍 MATLAB、DSP 及 FPGA 软硬件基本原理的基础上, 将通信系统的经典内容在 MATLAB、CCS 和 Quartus 三大主流软件平台上分别进行了仿真开发, 并相互验证结果的正确性。本书内容丰富, 实用性强, 按照“基本理论→MATLAB 实现→DSP 实现→FPGA 实现”的流程, 多层次展现通信系统的开发手段, 并通过图解的方式引导读者快速进入通信系统开发的大门。

本书可作为电气信息类本科生与研究生的 MATLAB、DSP 与 FPGA 实验入门教材, 也可作为通信工程技术人员的培训教材或自学指导书。

### 图书在版编目 (CIP) 数据

通信系统仿真开发: 基于 MATLAB、DSP 及 FPGA 的设计与实现 / 沈良等编著. —北京: 机械工业出版社, 2017. 1

ISBN 978-7-111-56020-3

I. ①通… II. ①沈… III. ①通信系统—系统仿真 IV. ①TN914

中国版本图书馆 CIP 数据核字 (2017) 第 027090 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 李馨馨 责任编辑: 李馨馨

责任校对: 张艳霞 责任印制: 李 昂

三河市国英印务有限公司

2017 年 4 月第 1 版·第 1 次印刷

184mm×260mm·18.5 印张·452 千字

0001-2500 册

标准书号: ISBN 978-7-111-56020-3

定价: 49.80 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

电话服务

网络服务

服务咨询热线: 010-88379833

机工官网: [www.cmpbook.com](http://www.cmpbook.com)

读者购书热线: 010-88379649

机工官博: [weibo.com/cmp1952](http://weibo.com/cmp1952)

教育服务网: [www.cmpedu.com](http://www.cmpedu.com)

封面无防伪标均为盗版

金书网: [www.golden-book.com](http://www.golden-book.com)

# 前 言

近年来,随着微电子技术、数字信号处理技术的飞速发展,数字信号处理器(Digital Signal Processor, DSP)和现场可编程门阵列(Field Programmable Gate Array, FPGA)取得了巨大的进步,在处理速度、运算精度、处理器结构、指令系统、指令流程等诸多方面都有了较大提高,并迅速在语音、雷达、声纳、地震、图像、通信系统、系统控制、生物医学工程、遥感遥测、地质勘测、航空航天、电力系统、故障检测、自动化仪表等众多领域获得了极其广泛的应用。当前,以DSP、FPGA芯片及外围设备为主,正在形成一个具有较大潜力的产业和市场。

DSP芯片的主要供应商包括美国的德州仪器公司(TI)、AD公司、AT&T公司和Motorola公司等。其中,TI公司的DSP芯片占世界DSP芯片市场近50%,在国内也被广泛采用。FPGA的主要供应商包括美国的Altera和Xilinx公司。

目前,介绍DSP和FPGA原理与应用的图书很多,但是快速引导学生进行DSP和FPGA实验的图书并不多。为了填补这一空白,同时配合“通信系统综合实验”课程的开设,本书通过将MATLAB、DSP、FPGA原理与具体实验相结合,一步一步地引导学生进行综合实验,使其可以快速入门。为了配合书中的实验,辅以北京合众达公司开发的EED-DTK DSP实验箱以及杭州康芯电子公司的GW-48 FPGA实验箱。

本书主要特点如下:①内容可以满足DSP基本原理、数字信号处理、通信原理、现代电子系统设计和通信系统综合实验等课程的DSP与FPGA实验要求;②每一个实验都分成4个层次,即基本理论、MATLAB实验、DSP实验与FPGA实验,这4个层次充分体现了DSP与FPGA的开发过程;③采用“傻瓜”式引导方法,每个实验的重要步骤及实验结果通过图解的方法来讲解。

本书可作为通信工程技术人员和通信工程专业本科生与研究生的DSP与FPGA实验入门教材或自学指导书。

本书第1章由沈良编写,第2章由任国春编写,第3、4、8章由高瞻编写,第5章和第7章由童晓兵编写,第6章由张玉明编写。

由于编者水平有限,错误和疏漏之处在所难免,敬请广大读者批评指正。

编 者

# 目 录

## 前言

第 1 章 MATLAB 应用导论 .....	1	3.2 FPGA 系统设计 .....	48
1.1 MATLAB 概述 .....	1	3.2.1 FPGA 芯片的基本原理 .....	48
1.2 MATLAB 的使用 .....	1	3.2.2 FPGA 设计开发流程 .....	49
1.3 MATLAB 语言 .....	4	3.2.3 FPGA 与 CPLD 的比较 .....	51
1.3.1 MATLAB 变量和数值表示 .....	4	3.2.4 FPGA 与 DSP 的比较 .....	52
1.3.2 矩阵基础 .....	5	3.3 Quartus 的使用 .....	53
1.3.3 MATLAB 算术运算 .....	7	3.3.1 Quartus II 的特点 .....	53
1.3.4 MATLAB 关系和逻辑运算 .....	8	3.3.2 基于 Quartus II 的开发设计 流程 .....	53
1.3.5 MATLAB 程序设计结构 .....	9	3.3.3 基于 Quartus II 的 VHDL 设计 方法 .....	54
1.3.6 数学函数和库函数 .....	11	3.4 VHDL 语言 .....	63
1.3.7 MATLAB 绘图 .....	13	3.4.1 VHDL 语言的诞生 .....	63
1.3.8 MATLAB 数据的输入/输出 .....	16	3.4.2 VHDL 程序的基本结构 .....	64
1.4 MATLAB 语言编程实例 .....	17	3.4.3 VHDL 数据类型 .....	65
第 2 章 DSP 系统导论 .....	20	3.4.4 VHDL 的主要描述语句 .....	66
2.1 DSP 概述 .....	20	3.4.5 VHDL 的进程 .....	67
2.1.1 DSP 芯片及其特点 .....	20	3.5 VHDL 语言编程实例 .....	68
2.1.2 DSP 芯片的种类 .....	21	第 4 章 信源编译码 .....	71
2.1.3 DSP 芯片的应用 .....	22	4.1 信源编译码概述 .....	71
2.2 DSP 系统设计 .....	22	4.2 信源编译码的基本原理 .....	71
2.2.1 典型的 DSP 系统构成 .....	22	4.3 PCM 的设计实现 .....	72
2.2.2 DSP 系统设计过程 .....	24	4.3.1 13 折线 PCM 的 MATLAB 实现 .....	76
2.3 CCS 的使用 .....	25	4.3.2 13 折线 PCM 的 DSP 实现 .....	80
2.3.1 CCS 的安装及设置 .....	25	4.3.3 13 折线 PCM 的 FPGA 实现 .....	84
2.3.2 CCS 的窗口、菜单栏和工具栏 的介绍 .....	28	4.4 CVSD 的设计实现 .....	90
2.3.3 CCS 的调试与使用 .....	31	4.4.1 CVSD 的 MATLAB 实现 .....	92
2.4 DSP 的 C 语言开发 .....	37	4.4.2 CVSD 的 DSP 实现 .....	97
2.4.1 CCS 支持的 C 语言 .....	37	4.4.3 CVSD 的 FPGA 实现 .....	103
2.4.2 DSP 芯片开发的编程方法 .....	40	第 5 章 数字基带传输 .....	110
2.4.3 C 语言程序开发的过程 .....	42	5.1 数字基带信号传输系统 .....	110
2.5 DSP 的 C 语言编程实例 .....	44	5.2 发送滤波器的基本原理 .....	111
第 3 章 FPGA 系统导论 .....	48	5.3 数字基带传输的 MATLAB	

实现 .....	112	7.2.1 BPSK 调制的基本原理 .....	211
5.3.1 常用的 MATLAB 函数 .....	112	7.2.2 BPSK 调制的 MATLAB 实现 ...	212
5.3.2 脉冲成型的 MATLAB 实现 .....	118	7.2.3 BPSK 调制的 DSP 实现 .....	215
5.3.3 数字基带传输的 MATLAB 实现 .....	122	7.2.4 BPSK 调制的 FPGA 实现 .....	223
5.4 数字基带传输的 DSP 实现 .....	124	7.3 QPSK 调制的实现 .....	226
5.5 数字基带传输的 FPGA 实现 ...	129	7.3.1 QPSK 调制的基本原理 .....	226
<b>第 6 章 信道编译码</b> .....	143	7.3.2 QPSK 调制的 MATLAB 实现 ...	227
6.1 信道编译码概述 .....	143	7.3.3 QPSK 调制的 DSP 实现 .....	233
6.2 汉明码的基本原理 .....	143	7.3.4 QPSK 调制的 FPGA 实现 .....	246
6.2.1 分组码的定义 .....	143	7.4 QAM 调制的实现 .....	252
6.2.2 生成矩阵 $G$ 和监督矩阵 $H$ .....	144	7.4.1 QAM 调制的基本原理 .....	252
6.2.3 汉明码的编译码算法 .....	145	7.4.2 16QAM 调制的 MATLAB 实现 .....	253
6.3 卷积码的基本原理 .....	146	7.4.3 16QAM 调制的 DSP 实现 .....	257
6.3.1 卷积编码算法 .....	147	7.4.4 16QAM 调制的 FPGA 实现 .....	267
6.3.2 Viterbi 译码算法 .....	149	<b>第 8 章 个性化实验</b> .....	273
6.4 汉明码的设计 .....	152	8.1 自适应均衡的 DSP 实现 .....	273
6.4.1 汉明码的 MATLAB 实现 .....	152	8.1.1 自适应均衡的基本原理 .....	273
6.4.2 汉明码的 DSP 实现 .....	156	8.1.2 自适应均衡的设计方法 .....	275
6.4.3 汉明码的 FPGA 实现 .....	161	8.1.3 自适应均衡的 MATLAB 和 DSP 实现 .....	276
6.5 卷积码的设计 .....	179	8.2 直接序列扩频的 FPGA 实现 ...	284
6.5.1 卷积码的 MATLAB 实现 .....	179	8.2.1 直接序列扩频的基本原理 .....	284
6.5.2 卷积码的 DSP 实现 .....	185	8.2.2 直接序列扩频的设计方法 .....	285
6.5.3 卷积码的 FPGA 实现 .....	192	8.2.3 直接序列扩频的 MATLAB 和 FPGA 实现 .....	286
<b>第 7 章 数字调制解调</b> .....	211	<b>参考文献</b> .....	290
7.1 数字调制的分类 .....	211		
7.2 BPSK 调制的实现 .....	211		

# 第1章 MATLAB 应用导论

## 1.1 MATLAB 概述

MATLAB 的首创者是在数值线性代数领域颇有影响的 Cleve Moler 博士，他开发了 MATLAB 软件。Moler 博士等一批数学家和软件专家组建了 MathWorks 软件公司，专门从事 MATLAB 的改进与扩展。MATLAB 有以下特点：

- 1) MATLAB 以矩阵作为基本编程单元，使矩阵操作非常便捷。
- 2) MATLAB 语句书写简单，语法也不复杂，各种表达式书写如同在草稿纸上演算一样，与人们的手工运算相近，容易使用。
- 3) MATLAB 语句功能强大，一条语句往往相当于其他高级语言中的几十条、甚至上百条。例如，C 语言 FFT 子程序有 70 多行，而 MATLAB 只需一个 `fft` 函数即可实现对序列的 FFT 计算。所以，将其用于数字信号处理实验，可以大大提高实验效率。
- 4) MATLAB 系统具有丰富的图形功能。MATLAB 具有良好的用户界面，而且提供了丰富的图形显示函数，可满足各行业人员直观、方便地进行分析、计算和设计工作。
- 5) MATLAB 提供了许多面向应用问题求解的工具箱 (Toolbox) 函数，从而大大方便了各个领域的研究需要。例如，信号处理工具箱 (Signal Processing Toolbox) 就提供了大量的信号处理函数。

通信系统中的数字信号处理概念比较抽象，而且其数值计算相对烦琐，非常适合用 MATLAB 来进行研究和计算。现在 MATLAB 已经成为世界范围内公认的解决数字信号处理问题的标准软件。

## 1.2 MATLAB 的使用

### 1. 窗口说明

启动 MATLAB 软件 (以 MATLAB 6. x 为例)，会弹出如图 1-1 所示的窗口。该窗口通常称为 MATLAB 主窗口。在主窗口中，选择 View 菜单中的选项，可以打开以下子窗口：

- 1) 在命令窗口 (Command Window) 下可以直接输入 MATLAB 命令行并执行。命令窗口中的符号 “>>” 表示该行是一条命令，而没有该符号的行显示的是命令执行后的结果。
- 2) 命令历史窗口 (Command History) 显示曾经在命令窗口中输入过的命令。
- 3) 当前路径窗口 (Current Directory) 显示当前工作目录下的文件。
- 4) 工作空间窗口 (Workspace) 显示变量信息，如变量的当前值等。
- 5) 帮助窗口 (Help) 显示帮助文档。MATLAB 的帮助文档中提供了详细的使用说明，尤其是函数说明都辅以相应的例子进行解说。帮助窗口是学习和使用 MATLAB 的常用工具。

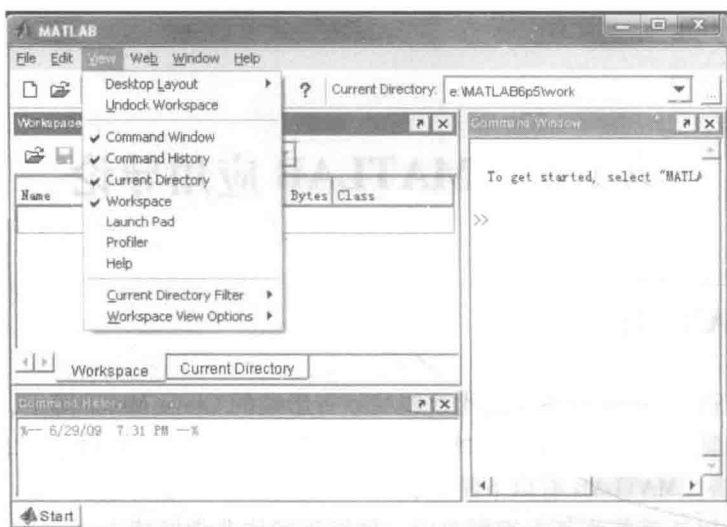


图 1-1 MATLAB 6. x 的主窗口

在命令窗口中直接输入“help 函数名”，可以得到函数的简要说明。例如，输入“help sin”，可以得到正弦函数使用的简要说明。

6) 编辑器窗口 (Editor) 对应的是 MATLAB 软件自带的文档编辑器窗口。该窗口是独立于主窗口之外的，用户可以编辑、调试自己的 M 文件。在不同版本的 MATLAB 下激活 Editor 编辑器的方法略有区别，在 Matlab 6. x 主窗口的菜单栏 File 中，通过单击 New (新建文件) 或 Open (打开文件) 激活 Editor 窗口。在 Matlab 7. x 版本中，可以单击 Desktop→Editor 命令，激活 Editor 窗口，如图 1-2 所示。MATLAB 6. x 和 MATLAB 7. x 的编辑器窗口分别如图 1-3 和图 1-4 所示。

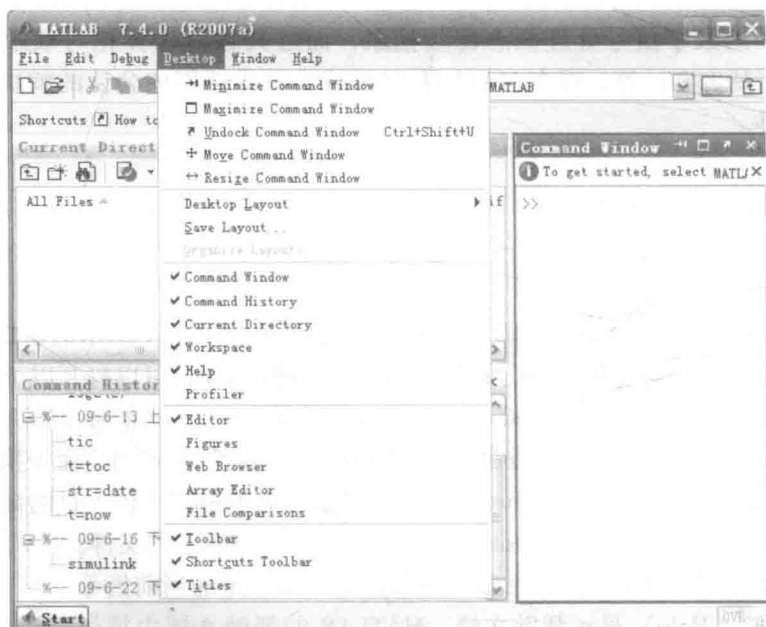


图 1-2 MATLAB 7. x 的主窗口



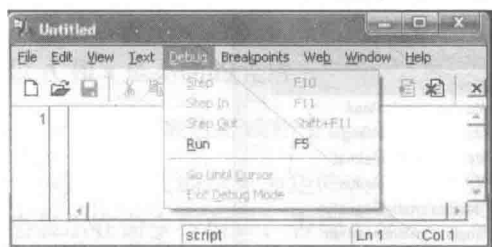


图 1-3 MATLAB 6. x 的编辑器窗口

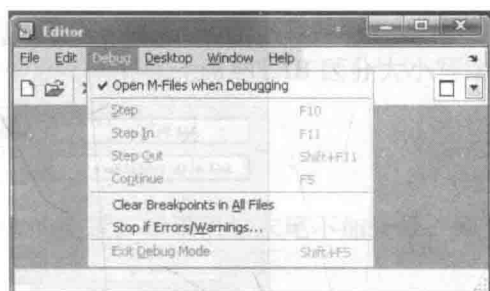


图 1-4 MATLAB 7. x 的编辑器窗口

在 MATLAB 中设置了一组对功能窗口进行操作的函数，可以在命令窗口或程序中使用。对功能窗口的常用操作函数见表 1-1。

表 1-1 对功能窗口的常用操作函数

函数名	功能	使用说明
clc	清屏，清除命令窗口下显示的内容	
clear	清除工作空间中存储的变量	
whos	查看所有变量特征或者指定变量特征	whos 或者 whos x
close	关闭当前绘图窗口或所有绘图窗口	close 或者 close all
edit	打开 M 文件编辑器	
exit	退出 MATLAB	

## 2. MATLAB 文件搜索路径设置

在 MATLAB 中有两类程序文件：一类是命令文件，另一类是函数文件，它们的扩展名都是“.m”，常称为 M 文件。命令文件中的语句一般通过一定的逻辑控制来实现特定的工作流程，还可能进行一些简单的计算，而工作流程中一些通用的计算或处理一般则由函数文件所描述的函数来完成。MATLAB 的函数输入参数可以是多个，函数执行后的返回值也可以是多个，这与很多高级语言是不同的。

由于 MATLAB 是通过搜索路径来寻找 M 文件并执行的，因此 MATLAB 的系统文件、工具箱函数以及用户自己编写的文件都应在搜索路径之内，所有不在此搜索路径中的文件是无法执行的。下面详细介绍将文件添加到搜索路径的方法（以 MATLAB 7. x 版本为例）。

1) 在磁盘建立一个文件夹，用来存放编好的程序。例如，在 D 盘新建一个目录，名称可以用姓名的拼音加学号后 4 位（如 zhangsan1234），建好以后，该文件夹的路径为 D:\zhangsan1234。

2) 在菜单栏中单击 File→Set Path 命令，弹出如图 1-5 所示的窗口。

3) 单击 Add Folder 按钮，会弹出一个“浏览文件夹”窗口，选中 M 文件存放的文件夹，单击“确定”按钮后浏览文件夹窗口消失，此时必须单击 Save 按钮，这样刚才指定的文件夹路径才会存入 MATLAB 默认的搜索路径。

4) 单击 Add with Subfolders 按钮，将指定文件夹内的所有子文件夹也加入到 MATLAB 默认的搜索路径中，单击 Save 按钮保存设置。当用户将文件分开存放，如将命令文件和函

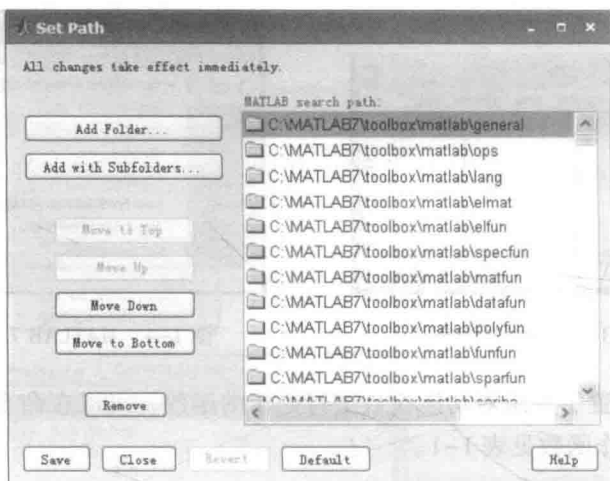


图 1-5 MATLAB 7. x 的设置路径窗口

数文件分别存放在同一文件夹下的不同子文件夹中时，单击 Add with Subfolders 按钮是很实用的方法。

### 3. 程序的编写和调试

M 文件可以在任何文本编辑器下编写，一般建议在 MATLAB 自带的编辑器下编写，以方便调试。注意，要想在 Editor 编辑器中调试程序，需要在 MATLAB 主窗口中激活 Editor 窗口。

M 文件的文件名由字母、数字和下画线构成，并且必须以字母开头，中间不能加空格，不能使用中文字符。函数文件的名称必须和函数名相同。

在 Editor 窗口的菜单栏中单击 Debug→Run 命令即可执行程序。另外，F5 键是执行程序的快捷键。在 Debug 或者 Breakpoints 下拉菜单中，可以看到有很多辅助用户调试的选项，如 Step（单步执行）、Set/Clear BreakPoint（增加/清除断点）等。如果程序运行出错，在命令行窗口会出现红色的文字提示，单击提示文字，则光标会定义到出错的程序代码，极大地方便了用户调试。最常见的错误有冒号和分号混淆，在切换到中文输入法后输入了中文标点符号，拼写错误等。

## 1.3 MATLAB 语言

### 1.3.1 MATLAB 变量和数值表示

MATLAB 是高级的矩阵/阵列语言，它具有控制流向语句、函数、数据结构、输入/输出及面向对象编程等特色。它既适用于可立即得到结果的小程序编程，又适用于完整求解复杂应用问题的大程序编程。

#### 1. 变量

MATLAB 最基本、最重要的功能就是进行实数或复数的矩阵运算。MATLAB 的基本变量均代表一个矩阵，向量和标量均作为特殊的矩阵来处理，可以很方便地进行向量和标量的

运算。矩阵一般由语句和函数产生，也可以从外部的数据文件读入。

变量名由字母、数字或下画线构成，并且必须以字母开头。MATLAB 区分大小写字母，所以 A 和 a 是不同的变量。对变量的赋值一般采用赋值语句：

变量 = 表达式 或者 变量 = 表达式；

一般表达式的结果为矩阵，它赋值给等号左边的变量。如果语句末尾不加分号，则变量值会显示在命令行窗口，否则不显示。

MATLAB 中提供了一些固定变量，如 ans、pi、Inf、NaN，一般用户给变量取名字时，不要跟这些名称一样，否则用户自定义的变量值会取代固定变量的值，易导致不容易检查出来的错误。常用的固定变量有以下 4 种。

1) ans: 在没有定义变量名时，系统默认变量名为 ans。例如，在命令行窗口输入 3，按〈Enter〉键，则屏幕显示 ans = 3。

2) pi: 变量 pi 就是数学上的  $\pi$ 。

3) Inf: 变量 Inf 表示无穷大，当程序中出现除数为 0 时，得到结果就是 Inf。

4) NaN: 变量 NaN 表示不确定值，它由 Inf/Inf 或者 0/0 的运算产生。

## 2. 数值

MATLAB 中采用十进制数，并且可以用科学计数法表示很大的数或者很小的数，如  $1e6$  就是十进制的 1000000， $1e-6$  就是十进制的 0.000001。

i 或 j 是虚数符号， $3+4i$  和  $3+4j$  表示同一个复数。注意虚数符号必须写在实数之后。i 或 j 也可以作为一般变量使用，系统能够自动识别它是虚数符号，还是一般变量。例如，在没有给 i 和 j 赋值的情况下，在命令行窗口输入 i-j，则显示结果为 0。

### 1.3.2 矩阵基础

#### 1. 矩阵产生（变量产生）

MATLAB 的基本变量均代表一个矩阵。通常采用输入元素列表或者利用各种函数产生矩阵。

##### (1) 输入元素列表法

输入元素列表时，按下列约定输入：

- 矩阵中所有元素用方括号括起来。
- 矩阵行中的元素以空格或逗号间隔。
- 矩阵行与行之间用分号或者回车间隔。

例如，在命令窗口输入

```
a = [1,2,3;4,5,6;7,8,9]
```

按〈Enter〉键执行后，则显示

```
a =  
    1    2    3  
    4    5    6  
    7    8    9
```

利用冒号操作符可以更便捷地产生矩阵。例如， $b = [0:2:10]$ 表示从0开始，以2为步长（步长为1可省略），一直递增到10，产生一个矩阵，该语句执行的结果为

```
b =  
    0    2    4    6    8   10
```

## (2) 函数法

MATLAB中的矩阵可以通过标准的M文件函数产生，下面介绍一些常用的矩阵函数。

1) ones: 产生元素全1矩阵函数。ones(N)产生一个 $N \times N$ 的全1矩阵，ones(M,N)产生一个 $M \times N$ 的全1矩阵。

例如，产生2行2列的矩阵。在命令窗口输入 $a = \text{ones}(2)$ ，执行结果为

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

再如，产生一个2行4列的全1矩阵。在命令窗口输入 $a = \text{ones}(2,4)$ ，执行结果为

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

2) zeros: 产生全0的矩阵。用法与ones一致，只是产生的矩阵元素都是0。

3) eye: 产生单位矩阵。

例如，产生 $3 \times 3$ 的单位矩阵。在命令窗口输入 $a = \text{eye}(3)$ ，执行结果为

```
a =  
    1    0    0  
    0    1    0  
    0    0    1
```

4) rand: 产生(0, 1)之间服从均匀分布的随机矩阵。

5) randn: 产生服从均值为0、方差为1的正态分布的随机矩阵。

## 2. 矩阵下标

矩阵中的每一个元素可以用括号中的下标表示， $a(i,j)$ 表示矩阵中处于第*i*行第*j*列的元素。利用冒号，可以表示一个向量或子矩阵， $a(:,j)$ 表示第*j*列元素组成的向量， $a(i,:)$ 表示第*i*行元素组成的向量，而 $a(i:j,k:m)$ 则表示一个 $(j-i+1) \times (m-k+1)$ 的子矩阵。例如，如果

```
a =  
    1    2    3  
    4    5    6  
    7    8    9
```

则， $a(1,3) =$

```
3
```

$a(:,3) =$

```

3
6
9
a(3,:) =
7 8 9
a(2:3,2:3) =
5 6
8 9

```

### 3. 矩阵转置

常用的矩阵转置操作符见表 1-2。

表 1-2 矩阵转置操作符

'	矩阵共轭转置	'	矩阵转置
---	--------	---	------

例如: a =

```

1 + 2i    3 + 4i
5 + 6i    7 + 8i

```

则在命令窗口输入 a' 得到

```

ans =
1 - 2i    5 - 6i
3 - 4i    7 - 8i

```

则在命令窗口输入 a.' 得到

```

ans =
1 + 2i    5 + 6i
3 + 4i    7 + 8i

```

### 1.3.3 MATLAB 算术运算

MATLAB 基本的算术运算操作符见表 1-3。

表 1-3 基本算术运算操作符

运算符	说 明	运算符	说 明
+	加法	.*	点加, 矩阵元素对应相加, 与加法效果相同
-	减法	.-	点减, 矩阵元素对应相减, 与减法效果相同
*	乘法	.*	点乘, 矩阵对应元素相乘
/	除法	./	点除, 矩阵对应元素相除
\	左除	.\	点左除, 矩阵对应元素左除
^	乘方	.^	点乘方, 矩阵对应元素乘方

MATLAB 的算术运算分为矩阵运算和矩阵数量运算。进行运算的矩阵其大小必须严格符合线性代数的规则。用点符号“.”(即句点号)来区分矩阵运算和矩阵数量运算。对于加法和减法运算而言, 两者的运算规则是相同的, 所以没有点加“.\*”和点减“.-”。

## 1. 加法和减法

$A + B$  和  $A - B$  是最简单的算术运算，其中  $A$  与  $B$  应该具有相同的维数。

## 2. 乘法和点乘

1) 矩阵乘法， $C = A * B$ ，完成矩阵  $A$ 、 $B$  的线性代数积，即

$$C(i,j) = \sum_{k=1}^n A(i,k)B(k,j)$$

2) 矩阵数量乘法， $C = A .* B$ ，完成  $A$ 、 $B$  的对应元素相乘，即  $C(i,j) = A(i,j)B(i,j)$ ， $A$  如果是 1 维矢量（即标量），则“.”可省略。

要注意的是，矩阵的乘法和点乘的结果是不同的。例如  $A = [1,2]$ ， $B = [3,4]$ ，则  $A * B = [3,8]$ ，因为此点乘是对应元素相乘；而  $A * B' = 11$ ，因为此乘法是矩阵相乘。

## 3. 除法和点除

1) 除法。除法分为右除和左除。

- 右除。 $A/B$  完成矩阵的右除，相当于  $A$  乘  $B$  的逆阵，即  $A * \text{inv}(B)$ ， $B$  必须是方阵。

- 左除。 $A \setminus B$  完成矩阵的左除，相当于  $A$  的逆阵乘  $B$ ，即  $\text{inv}(A) * B$ ， $A$  必须是方阵。

2) 点除。点除分为点右除和点左除。

- 点右除。 $C = A ./ B$  完成矩阵  $A$ 、 $B$  对应元素相除的运算，即  $C(i,j) = A(i,j)/B(i,j)$ 。

- 点左除。 $C = A . \setminus B$  完成矩阵  $A$ 、 $B$  对应元素左除的运算，即  $C(i,j) = B(i,j)/A(i,j)$ 。

## 4. 乘方和点乘方

(1) 乘方

- $p$  为标量， $X$  为矩阵，则  $X^p$  为计算矩阵  $X$  的  $p$  次幂。

- $x$  为标量， $P$  为矩阵，则  $x^P$  的计算用到矩阵  $P$  的特征值和特征向量，不是一般意义上的乘方运算。

- $X$ 、 $P$  都为矩阵时， $X^P$  的操作无法求解，会提示语法错误。

(2) 点乘方

$C = A.^B$  是矩阵元素对元素的乘方，即  $C(i,j) = A(i,j).^B(i,j)$ 。

## 1.3.4 MATLAB 关系和逻辑运算

### 1. 关系运算

在 MATLAB 中，有 6 个关系操作符，见表 1-4。

表 1-4 关系操作符

关系操作符	含义
<	小于
<=	小于或等于
>	大于
>=	大于或等于
==	等于
~=	不等于

若关系运算的结果是 1，则表明为真；若关系运算的结果是 0，则表明为假。

例如： $A = [1,2;4,6]$

```
B = [2,4;7,3]
```

```
C = (A > B)
```

则输出结果为

```
C =  
0 0  
0 1
```

## 2. 逻辑运算

在 MATLAB 中，有 3 种逻辑运算符，见表 1-5。

表 1-5 逻辑运算符

逻辑运算符	含 义
&	与
	或
~	非

若逻辑运算的结果是 1，则表明为真；若逻辑运算的结果是 0，则表明为假。

例如：a = 12;

```
b = 1;
```

```
if(a > 10) & (b == 1)
```

```
    a = a + 3;
```

```
else
```

```
    a = a - 3;
```

```
end
```

则输出结果为

```
a =  
15
```

### 1.3.5 MATLAB 程序设计结构

MATLAB 与其他大部分计算机高级语言一样，有自己的设计结构。设计结构使 MATLAB 远远超出桌面计算器的范畴，使之成为一种高水平的矩阵计算语言，并得到了广泛应用。它的主要程序设计结构包括顺序结构、条件结构和循环结构 3 种。

#### 1. 顺序结构

顺序结构是指按照程序中语句的排列顺序依次执行，直到程序的最后一条语句。

#### 2. 条件结构

条件结构是指根据给定的条件成立或不成立，分别执行不同的语句。MATLAB 用于实现条件结构的常见语句有 if 语句和 switch 语句。

##### (1) if 语句

在 MATLAB 中，if 语句有以下 3 种格式。

1) 单分支 if 语句。

语句格式：

```
if 条件
    语句组
end
```

含义：当条件成立时，执行条件下面的语句组。

2) 双分支 if 语句。

语句格式：

```
if 条件
    语句组 1
else
    语句组 2
end
```

含义：当条件成立时，执行语句组 1，否则执行语句组 2。

3) 多分支 if 语句。

语句格式：

```
if 条件 1
    语句组 1
elseif 条件 2
    语句组 2
...
elseif 条件 m
    语句组 m
else
    语句组 n
end
```

含义：当条件 1 成立时，执行语句组 1；当条件 2 成立时，执行语句组 2……否则执行语句组  $n$ 。

(2) switch 语句

switch 语句根据变量或表达式的取值不同，分别执行不同的语句，其语句格式如下：

```
switch 表达式或变量
case 值 1
    语句组 1
case 值 2
    语句组 2
...
case 值 m
    语句组 m
otherwise
    语句组 n
end
```



当表达式或变量满足值 1 时，执行语句组 1；当表达式或变量满足值 2 时，执行语句组 2……否则执行语句组  $n$ 。

### 3. 循环结构

循环结构是指按照给定的条件，重复执行指定的语句。MATLAB 实现循环结构的常见语句有 for 语句和 while 语句。

#### (1) for 语句

for 语句的格式如下：

```
for 循环变量 = 表达式 1:表达式 2:表达式 3
    循环体语句
end
```

其中，表达式 1 的值为循环变量的初值，表达式 2 的值为步长，表达式 3 的值为循环变量的终值。步长为 1 时，表达式 2 可以省略。

#### (2) while 语句

while 语句的一般格式如下：

```
while 条件
    循环体语句
end
```

其执行过程如下：若条件成立，则执行循环体语句。执行后再判断条件是否成立，如果条件仍然成立，则执行循环体语句；如果不成立，则跳出循环。

## 1.3.6 数学函数和库函数

### 1. 数学函数

MATLAB 提供了一些基本数学函数，如正弦、余弦函数，也提供了一些特殊的数学函数，如贝塞尔函数。表 1-6 给出了 MATLAB 中常用的数学函数。

表 1-6 MATLAB 中常用的数学函数

数学函数	含 义	数学函数	含 义
abs(x)	求绝对值或复数的模	log(x)	以 e 为底的自然对数
acos(x)	反余弦函数	log10(x)	以 10 为底的常用对数
asin(x)	反正弦函数	max(x)	求最大值
atan(x)	反正切函数	min(x)	求最小值
ceil(x)	向正无穷大方向取整	real(x)	求复数的实部
conj(x)	求共轭复数	rem(x,y)	除法后求余数
cos(x)	余弦函数	round(x)	四舍五入取整
exp(x)	指数函数	sign(x)	符号函数
fix(x)	向零方向取整	sin(x)	正弦函数
floor(x)	向负无穷大方向取整	sqrt(x)	求平方根
imag(x)	求复数的虚部	tan(x)	正切函数